WESTERN SYDNEY UNIVERSITY

Cognitive Radio Networks: Quality of Service Considerations and Enhancements

Nabil Giweli

A thesis submitted in fulfilment for the degree of Doctor of Philosophy

School of Computing, Engineering and Mathematics

Western Sydney University

October 2018

Statement of Authentication

The work presented in this thesis is, to the best of my knowledge and belief, original except as acknowledged in the text. I hereby declare that I have not submitted this material, either in full or in part, for a degree at this or any other institution.

[redacted]

Nabil Giweli

October 2018

Copyright

Acknowledgements

I owe my deepest gratitude to my supervisor, Dr Seyed Sharhestani, for his encouragement, guidance and support. If it was not for the motivation and the guidance of Dr Seyed, completing this thesis would not have been possible. His suggestions helped me improve my problem solving capabilities and research skills.

Thanks are further due to my Co-supervisor Dr Hon Cheung, for his help in completing this thesis and for his continuing advice. I also thank him for providing me with valuable suggestions and comments during my work.

I owe a lot to my colleagues in the Networking, Security, and Cloud Research group for encouragement and valuable discussions during this work.

I warmly thank the administration and technical staff in the School of Computing, Engineering and Mathematics for their care and support during my candidature. I highly appreciate the workshops and other types of supporting that the Graduate Research School offers for higher degree research students.

This thesis has been professionally edited by Dr Margaret Johnson of The Book Doctor, in accordance with the guidelines established by the Institute of Professional Editors and the Deans and Directors of Graduate Studies.

Finally, I would like to thank my parents and family for all the sacrifices they have made for me, throughout my life and particularly during the completion of this thesis.

Contents

Tables

Figures

Abstract

The explosive growth of wireless and mobile networks, such as the Internet of Things and 5G, has led to a massive number of devices that primarily use wireless channels within a limited range of the radio frequency spectrum (RFS). The use of RFS is heavily regulated, both nationally and internationally, and is divided into licensed and unlicensed bands. While many of the licensed wireless bands are underutilised, useable unlicensed bands are usually overcrowded, making the efficient use of RFS one of the critical challenges faced by future wireless communication technologies. The cognitive radio (CR) concept is proposed as a promising solution for the underutilisation of useful RFS bands. Fundamentally, CR technology is based on determining the unoccupied licensed RFS bands, called spectrum white spaces or holes, and accessing them to achieve better RFS utilisation and transmission propagation. The holes are the frequencies unused by the licensed user, or primary user (PU). Based on spectrum sensing, a CR node, or secondary user (SU), senses the surrounding spectrum periodically to detect any potential PU transmission in the current channel and to identify the available spectrum holes. Under current RFS regulations, SUs may use spectrum holes as long as their transmissions do not interfere with those of the PU. However, effective spectrum sensing can introduce overheads to a CR node operation. Such overheads affect the quality of service (QoS) of the running applications. Reducing the sensing impact on the QoS is one of the key challenges to adopting CR technology, and more studies of QoS issues related to implementing CR features are needed. This thesis aims to address these QoS issues in CR while considered the enhancement of RFS utilisation.

This study concentrates on the spectrum sensing function, among other CR functions, because of its major impact on QoS and spectrum utilisation. Several spectrum sensing methods are reviewed to identify potential research gaps in analysing and addressing related QoS implications. It has been found that none of the well-known sensing techniques is suitable for all the diverse QoS requirements and RFS conditions: in fact, higher accuracy sensing methods cause a significant QoS degradation, as illustrated by several simulations in this work. For instance, QoS degradation caused by high-accuracy sensing has not yet been addressed in the IEEE 802.11e QoS mechanism used in the proposed CR standard, IEEE 802.11af (or White-Fi). This study finds that most of the strategies proposed to conduct

sensing are based on a fixed sensing method that is not adaptable to the changeable nature of QoS requirements. In contrast, this work confirms the necessity of using various sensing techniques and parameters during a CR node operation for better performance.

Accordingly, this thesis proposes a fuzzy logic decision-making mechanism for selecting the proper sensing method in real-time spectrum sensing. The selection is based on several identified operation requirements, QoS in particular, and other constraints. The fuzzy inference system (FIS) is developed in MATLAB to implement the proposed selection mechanism and to demonstrate the logical selection of the FIS based on changing requirements. Because of its flexibility and simplicity, the FIS can be executed in CR nodes with minimal overhead on their computational resources, so it can be deployed in various CR technologies and under different sensing strategies. The sensing strategy in this research refers to the way of determining when, for how long, and what to sense in real-time situations. To improve performance, the medium access control (MAC) protocol used in the communication channel must be considered in designing the sensing strategy. In this research, a sensing strategy called QoS awareness CR MAC (QACR-MAC), is proposed to integrate with IEEE 802.11e to enhance QoS in White-Fi networks. A CR node based on IEEE82.11 standards is implemented in the defacto simulation tool Riverbed Modeler (formerly known as OPNET) to use different sensing strategies. In particular, the CR node is designed to use the proposed selection mechanism that has been implemented in MATLAB within QACR-MAC. Then extensive simulation scenarios are designed and conducted to study the impact of sensing duration and accuracy on the QoS and evaluate the proposed solutions.

Simulation results illustrate that using a sensing method based on a fuzzy logic decision-making mechanism, as proposed in this thesis, outperforms fixed sensing strategies when sensing accuracy and QoS are concerned. The results of evaluating QACR-MAC under different MAC protocols settings demonstrate that this proposed strategy can help improve QoS significantly in White-Fi networks. To the best of my knowledge, QACR-MAC is the first sensing strategy proposed and implemented to maintain IEEE 802.11e QoS features when high-accuracy sensing methods are utilised. Future work will study if a cooperative sensing concept used in QACR-MAC could also minimise overhead and give better performance.

Publications

Part of the material in this thesis has been published as follows:

- **N. Giweli, S. Shahrestani, and H. Cheung, 'Spectrum Sensing Strategy to Enhance the QoS in White-Fi Networks,' International Journal of Computer Networks & Communications (IJCNC), vol. 10, pp. 1-12, 2018.**

- **N Giweli, S Shahrestani, H Cheung, 'Spectrum Sensing Approach Based on QoS Requirements in White-Fi Networks', 4th International Conference on Wireless and Mobile Network (WiMNET 2017), Sydney, Australia, 2017, pp. 91-100**

- **N. Giweli, S. A. Shahrestani, and H. N. Cheung, 'Cognitive radio with spectrum sensing for future networks,' in Networks of the Future: Architectures, Technologies, and Implementations, ed. Chapman and Hall/CRC, 2017, pp. 3-23.**

- **N. Giweli, S. Shahrestani, and H. Cheung, "QoS-aware spectrum sensing in White-Fi cognitive radio networks," in 2016 10th International Conference on Signal Processing and Communication Systems (ICSPCS), Gold Coast, QLD, Australia, 2016, pp. 1-8.**

- **N. Giweli, S. Shahrestani, and H. Cheung, 'Selection of spectrum sensing method to enhance Qos in cognitive radio networks,' I*nternational Journal of Wireless & Mobile Networks* (IJWMN), vol. 8, pp. 39-50, 2016.**

- **N. Giweli, S. Shahrestani, and H. Cheung, 'Selecting the sensing method in cognitive radio and future networks: a QoS-Aware fuzzy scheme,' in 2015 IEEE International Conference on Data Science and Data Intensive Systems, Sydney, Australia, 2015, pp. 497-504.**

- **N. Giweli, S. Shahrestani, and H. Cheung, 'Spectrum sensing in cognitive radio networks: QoS considerations,' in *Computer Science & Information Technology* (CS & IT), Proceedings of the Seventh International Conference on Network and Communications (NETCOM 2015), 26-27 December 2015, Sydney, Australia, 2015, vol. 5, no. 16, pp. 9-19.**

Abbreviations and Notations

AC	access category	
AC_BE	best-efforts access category	
AC_BK	background access category	
AC_VI	video access category	
AC_VO	voice access category	
AIFS	arbitration inter-frame space	
A-MPDU	aggregated-MAC protocol data unit	
A-MSDU	aggregated-MAC service data unit	
AP	access point	
aPPDUMaxTime	maximum time duration for transmitting a PPDU	
AWGN	additive white Gaussian noise	
B-ACK	block-ACKnowledgement	
BSS	basic service set	
CCA	clear channel assessment	
CCC	common control channel	
CR	cognitive radio	
CR-CSMA	cognitive radio- carrier sense multiple access	
CSMA/CA	carrier sense multiple access/collision avoidance	
CTS	clear to send	
CW	contention window	
DCF	distributed coordination function	
DIFS	DCF inter-frame space	
DSA	dynamic spectrum access	
DSSS	direct sequence spread spectrum	
ED	energy detection	
EDCA	enhanced distributed channel access	
EIFS	extended inter-frame space	
ETSI	European Telecommunications Standards Institute	
FCC	Federal Communications Commission	
FHSS	frequency-hopping spread spectrum	
FIS	fuzzy inference system	
GDB	geolocation database	
HCCA	HCF controlled channel access	
HCF	hybrid coordination function	
HT	high-throughput	
HT-STA	high throughput-station	
IEEE	Institute of Electrical and Electronics Engineers	
IR	infrared radiation	
ISM	industrial, scientific and medical	

MAC	medium access control	
MF	membership function	
MIMO	multi-input multi-output	
MPDU	MAC protocol data unit	
MSDU	MAC service data unit	
NAV	network allocation vector	
N_{op}	number of operational channels	
Ns	maximum number of channels to be sensed	
OFDM	orthogonal frequency division multiplexing	
OMF-MAC	opportunistic matched filter-based MAC	
OSI	open systems interconnection	
PCF	point coordination function	
P_d	probability of positive detection	
P_f	probability of false alarm detection	
P_m	probability of missed detection	
PPDU	physical protocol data unit	
PSDU	physical service data unit	
PU	primary user	
QACR-MAC	QoS awareness MAC protocol	
QC MAC	QoS-aware cognitive MAC	
QoS	quality of service	
RF	radio frequency	
RFS	radio frequency spectrum	
ROC	receiver operating characteristics	
RTP	real-time transport protocol	
RTS	request to send	
RxTx	receive transmit	
S_d	sensing duration variable	
S_{dMAX}	maximum sensing duration	
S_{dMIN}	minimum sensing duration	
SDR	software defined radio	
SNR	signal-to-noise ratio	
SU	secondary user	
T_{MaxTx}	maximum transmission time	
$T_{MaxWait}$	maximum waiting time	
TVHT	television very high throughput	
TXOP	transmission opportunity	
VHT	very high throughput	
WLAN	wireless local area network	

Chapter 1. Introduction

This chapter introduces this thesis. Background about the cognitive radio topic is introduced in Section 1.1 and details about the cognitive radio challenges and standardisation efforts are discussed in Section 1.2. The motivation for this work is represented in Section 1.3. An idea about the research questions addressed in this dissertation and their relevant scopes is illuminated in Section 1.4. The major findings and original contributions are listed in Section 1.5. Lastly, the thesis stages and outline are found in Section 1.6.

1.1 Introduction to cognitive radio (CR)

Typically, the radio frequency spectrum (RFS) is divided into frequency bands, regulated in most countries by their governments. Each country has a process for allocating the frequency bands, based on technical and economic factors. For instance, the Australian Communications and Media Authority is a federal government department responsible for RFS allocation in the Australian continent. The RFS allocation of Australia in 2013 is illustrated in Figure 1.1 [1].

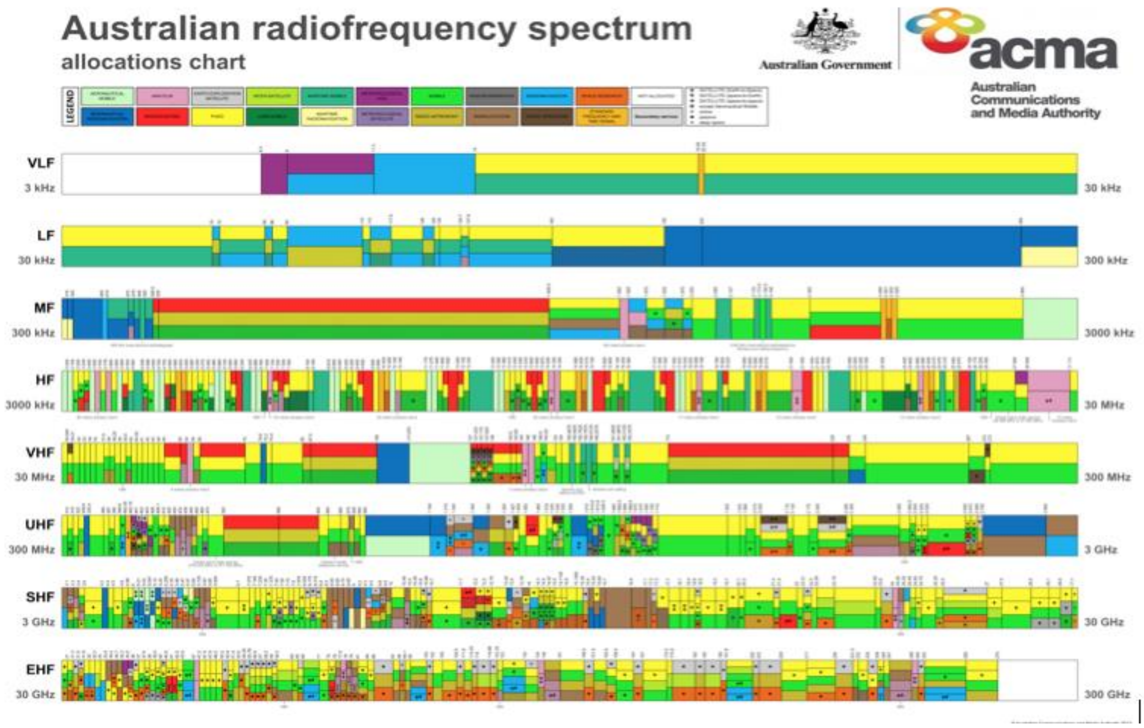


Figure 1.1 Australian radiofrequency spectrum allocations chart [1]

Although each country allocates its RFS independently, most governments do so in compliance with international and regional standards. For seamless worldwide interconnection, several organisations are responsible for standardising RFS allocation, including the International Telecommunication Union, European Telecommunications Standards Institute and the US Federal Communications Commission (FCC). In general, RFS bands are statically allocated as licensed or unlicensed, and the use of each licensed band is strictly assigned to a certain licensed user. An unlicensed band can be accessed and shared by anyone, with some regulatory constraints, so usable unlicensed bands are exposed to high demand as many wireless technologies and radio devices are designed to work in them. The Industrial, Scientific and Medical (ISM) bands are well-known examples that include the unlicensed frequency bands 902–928 MHz, 2400–2483.5 MHz, and 5725–5850 MHz. The ISM bands are widely used, especially by systems based on Wi-Fi and Bluetooth, under several specific regulations regarding operation requirements, such as power transmission and antenna gain [2].

As a result of RFS regulations, many licensed RFS bands are underutilised in terms of frequency, time and location. This has been found in measurements of frequency of occupancy conducted in several countries, including China [3], Europe [4], Singapore [5], New Zealand [6], South Africa [7], the United States [8] and Australia [9]. In the near future, current RFS regulations may not be able to handle the rapidly growing use of wireless communication technologies, with a widely expected increase in the demand for high transmission data rates. Therefore, the wireless research community and RFS regulatory organisations face the challenges of achieving high utilisation of the overall RFS and of overcoming the capacity scarcity in high-demand bands. Figure 1.2 shows the expected increase in spectrum demand from 2007 to 2020 in Australia for around 500-fold [10]. The Australian Communications and Media Authority has predicted a shortfall of the spectrum required to meet such massive demand. Overcoming these challenges will most likely lead to innovative wireless technologies and revision of regulations governing the use of RFS [11].

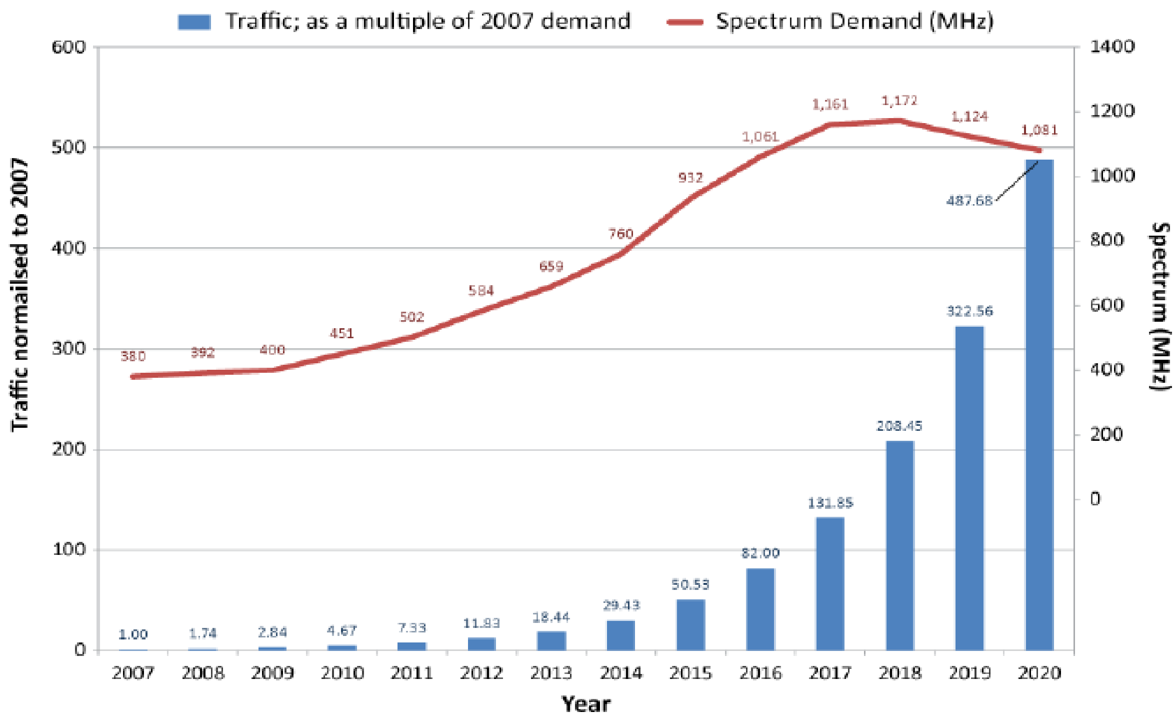


Figure 1.2 Expectations for spectrum demand and traffic to 2020 for Australia [10]

From a technical perspective, the concept of cognitive radio (CR), introduced in 1999 by Joseph Mitola [12], promises to achieve significant RFS utilisation and open new opportunistic uses of regulated RFS. Under CR, a radio device uses this technology, called a CR device, to allow an unlicensed user, called the secondary user (SU) or CR user, to use licensed RFS bands without harming the operations of the origin licensee (or primary user, PU). Given the underutilisation of the RFS, the licensed bands may have unoccupied periods, called holes or white spaces, depending on time, frequency or space. CR users can access these white spaces, in licensed as well as unlicensed bands, in real-time operations.

1.1.1. Cognitive radio definition

There are several proposed definitions for CR. The FCC offers a simple definition: ‘a radio that can change its transmitter parameters based on interaction with the environment in which it operates’ [13]. Other definitions provide more detail about features of the system, as shown in the list below [14-16]:

- **Awareness**: It is aware of its surrounding environment by a built-in sensing capability.

- **Intelligence**: It is fully programmable and consists of an intelligent wireless communication system that is capable of learning from and reasoning with the information gathered from its environment.
- **Adaptivity**: It can dynamically reconfigure its operation parameters, such as transmitting power, carrier frequency, networking protocols and modulation strategies so that it can adapt to variations in the RFS conditions and application requirements in a real-time.

These features are designed to achieve the main two objectives of the CR system [14]:

- highly reliable communication;
- efficient utilisation of RFS.

In the context of these two features, CR technology is projected to provide a universal radio platform that will be programmable to meet general-purpose applications for wireless systems [15]. This will become more feasible as progress is made in related fields of technology, particularly in radio equipment and software, digital signal processing, machine learning and wireless networking. Most of these technologies fall under one of the two pillar notions in the CR concept: the Software Defined Radio (SDR) and Dynamic Spectrum Access (DSA). The SDR concept, proposed by Joseph Mitola in the early nineties [17], was that one hardware platform may be reconfigured on the fly to support a wide range of wireless communication standards, frequency bands and technologies [18]. In contrast, the DSA technology exploits the SDR capability to allow wireless systems to access unused or less congested portions of the RFS without interfering with PU operations [19]. The word 'dynamic' in DSA refers to the paradigm shift in spectrum management from a static to a dynamic approach by which a wireless system can use different spectrum bands, including licensed bands [20]. The CR system combines the SDR and DSA capabilities to dynamically access the best spectrum band and reconfigure its operational parameters to best suit the current operating conditions [21].

1.1.2. CR functions and cycle

To achieve the objectives and features noted in the definitions above, several functions of the CR systems have to be executed logically, called the CR cycle. The first and most basic CR cycle was illustrated by Mitola [12] (see Figure 1.3). According to Mitola's model, the main functions of the CR are spectrum sensing, spectrum management/decision, spectrum sharing and spectrum mobility. More specifically, the CR system requires that an SU be able to:

- sense the surrounding spectrum to determine the spectrum holes (white spaces) and detect the PU's appearance;
- analyse and decide which spectrum hole is the most suitable to satisfy user/application requirements; i.e., quality of service (QoS) requirements;
- share the spectrum with CR neighbours, if any, with the best manageable fairness;
- seamlessly switch to another suitable spectrum hole to avoid harmful interference to the PU. Switching to another spectrum hole is referred to as spectrum handover or spectrum mobility [22].

It is an important requirement in the CR communication scenario that CR users will not cause problems or QoS degradation to PU operations, and this may result in a sequence of handovers between spectrum holes during communication, as shown in Figure 1.4. The handover has to be seamless and with minimal impact on the QoS of the application running on the CR [23].

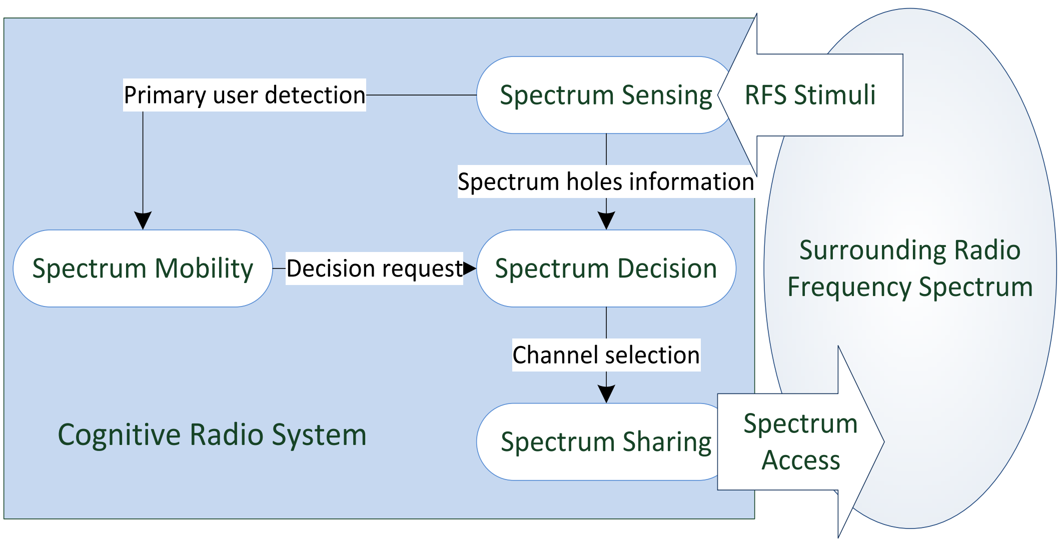


Figure 1.3 Basic CR cycle

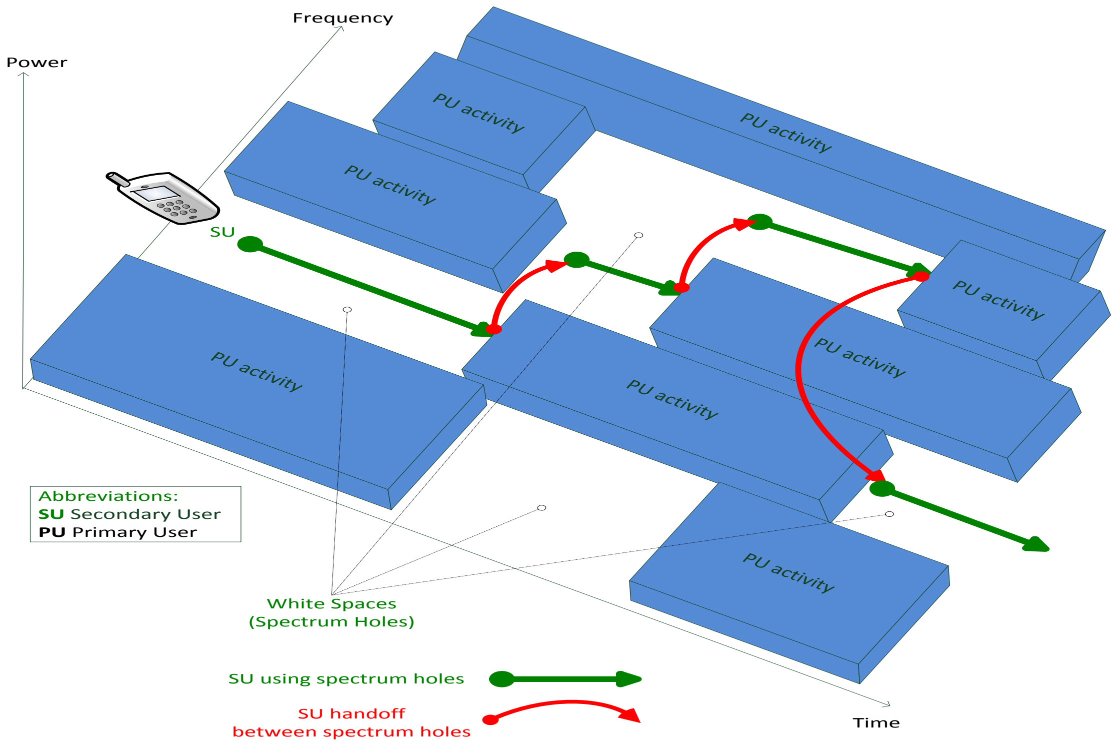


Figure 1.4 CR operation and handover

1.2. CR challenges and standardisation

CR technology is still in a development stage, and several technical challenges hinder its practical use. In addition, CR requires a business model that is suitable for the CR concept. This study is limited to technical aspects of the CR. In this section, several of the technical challenges related to CR functions and CR standardisation efforts are discussed.

1.2.1. Spectrum sensing and white space determination challenges

CR technology relies on what information can be gathered about the surrounding RFS. Detecting the PU's activities, mainly to find if the PU is using the spectrum band or not, is a major consideration in CR operation. This basic sensing operation requires sophisticated technologies that are accurate enough to avoid false alarms by mistakenly detecting a PU presence, and do not miss available holes [24]. In some cases, more than one PU may be using the same frequency band, so any detection approach must consider the possibility of multiple PU transmissions [25, 26]. Detecting SUs sharing the same hole is another advanced level of requirement that is needed when homogeneous SU networks compete for the same bands [27]. The nature of electromagnetic signals makes accurate sensing a complicated operation. Specifically, the signal-to-noise ratio (SNR) required for detection, the multipath fading of the PU signal, and the changing of noise/interference level with time and location, are major factors that may affect sensing accuracy [28, 29]. The hidden PU problem is another issue, if an SU cannot detect a PU when they are in the same transmission range because of problems caused by fading and shadowing of the signal [30]. Wideband sensing is an implementation challenge as it requires a high sampling rate for a very large radio frequency range, of a GHz or more, and high-speed signal processing devices [31]. The limited width of spectrum that a sensing technique can assess poses another constraint on how many channels can be sensed at a time.

Implementing a flexible sensing function is another challenge, as the characteristics of the radio environment and the various possible types of PU systems are rapidly changing [32]. Exchanging the sensing information between CR devices in a large network is expected to increase sensing accuracy, but how much CR radios can cooperate, and how much overhead is added by this cooperation, are still open questions [33]. Technically, a CR device cannot transmit during sensing, so the sensing operation should be as brief and as infrequent as possible without affecting its accuracy [32]. Alternative solutions, instead of sensing, have been proposed. For instance, so a CR device can determine the white space availability per location based on a central geolocation database (GDB) service either partially or entirely (in this case it is also called Senseless approach) [34, 35]. Although a GDB approach may help in the more efficient determination of white spaces, it raises several system and networking issues. For example, this approach will not be suitable for an un-centralised CR network [36]. It also requires a CR device constantly to provide its location, which consumes power and is an error-prone operation [35]. Other approaches that require the PU to cooperate, e.g., by sending beacons, are not preferred as they require a change in the PU operation and possibly its infrastructure [30]. Accurate sensing is essential as any imperfection may cause increasing error rates for both primary and secondary systems [37]. Notably, any experienced PU’s QoS degradation caused by CR technology will be a significant obstacle for practical application of a CR solution.

1.2.2. Spectrum management challenges

The cognition concept in CR is reflected from the learning and intelligence capacities that can be performed in spectrum management. The spectrum management function is dependent on the information provided about the spectrum by white space determination techniques, in particular, sensing techniques. As a consequence, all the challenges expected in white space determination will have an impact on the spectrum management decision. The QoS requirements of the running application have to be considered in the selection of the available spectrum holes [14]. For example, the selection of a channel that may have a high frequent use by a PU will reduce the QoS, particularly for delay-sensitive applications, as the CR device has to vacate the channel frequently. Such an issue shows the importance of considering the PU activity patterns of the target channel in the spectrum management algorithm [38].

When the idle portion of the spectrum is open to more than one SU, the activity pattern of the other SUs competing for that hole will also need to be considered [11]. In most cases, modelling such traffic patterns is challenging because of the random and unpredictable nature of PU and SU behaviours for many communication systems. In this context, providing a practical learning capability that can gradually improve the spectrum decision without the aid of external information is an essential challenge [39]. As seen in Figure 1.3, the spectrum management is the central function in CR architecture as proposed by Mitola [40], and has to deal with complex and multilayered issues relating to technical and regulatory policy requirements in real time to achieve spectral efficiency. Such a function requires capabilities in self-management, self-learning and self-reconfiguration that in turn need a set of algorithms that can provide these capabilities for a wide range of possible situations [41]. Achieving such capabilities requires an investigation of how to apply learning and reasoning algorithms and find related implementation issues [42]. For example, the game-theory has been used [43], as has a Bayesian network [44] and neural networks [45]. The available computation, time and power resources are the main constraints to implementing an intelligent spectrum management engine in a CR device. Also, the accuracy of a sensing operation will affect the effectiveness of the spectrum management.

1.2.3. Spectrum mobility challenges

In CR, an SU can share the spectrum with a PU in two proposed ways: overlay and underlay [21]. In overlay spectrum sharing, the SU has to vacate the spectrum band once the PU appears. In underlay spectrum sharing, the SU attempts to operate under a noise level acceptable to the PU. This means the CR network has to move from one spectrum band to another, whenever it is required, during its transmission. Typically, the handoff process is required to be executed with seamless data transmission and QoS assurance [46]. A similar challenge has been addressed in the vertical handover in Fourth Generation (4G) heterogeneous wireless networks, where several handoff algorithms are proposed [47]. However, the fluctuating nature of spectrum resources in CR networks raises several unique challenges that are not considered in these conventional algorithms [48]. In CR networks, available spectrum holes may not be contiguous, and may be found over a wide frequency range. Mobility between such holes leads to a switching delay much longer and more frequently than that in conventional wireless networks [23]. For seamless data transmission and QoS assurance, the mobility management in CR networks needs to deal with two issues: the dynamic availability of the spectrum holes and the heterogeneous nature of the CR network [48].

1.2.4. Spectrum sharing challenges

Technically, in the CR basic concept, an SU has the same priority for the use of an available spectrum hole as any other user, and so a hole that has optimal characteristics is a more attractive target for SUs competing in the same location and time. SUs sharing the available spectrum may be cooperative or non-cooperative [49]. In a non-cooperative scenario, in general, selfish behaviour is expected. In such a scenario, it is a challenge for each competing SU to avoid degradation of its performance caused by other competitors [50]. Such a problem will affect the competing SUs and may result in inefficient use of the spectrum [51]. In a cooperative scenario, SUs are willing and able to cooperate to achieve fair sharing of the available spectrum. However, this raised a challenge of how to exchange the information and control signals required for cooperation while minimising transmission and cooperation overheads [50]. Cooperation becomes more difficult when different CR systems are involved.

1.2.5. Other CR system challenges

In addition to the above-mentioned challenges related to the four CR functions, several other challenges are related to the upper layers of the communication protocol stack. The

CR functions are mainly performed at the medium access control (MAC) layer, which is a sublayer of the data link layer, and the physical layers of the open systems interconnection model. However, the protocols running at the application, transport and network layers are also affected by the CR functions. The basic model of a CR device is shown in Figure 1.5. For instance, the routing protocol has an important role in CR ad-hoc networks (CRAHNs) and routing protocols proposed for conventional ad-hoc wireless networks are not suitable for the new characteristics of CRAHNs [52]. The typical routing protocols should be modified to adapt to the dynamic spectrum behaviour of CRAHNs [53, 54]. The TCP protocol also needs to be adapted to the CR systems as its congestion control mechanism is negatively affected by sensing and mobility operations [55, 56].

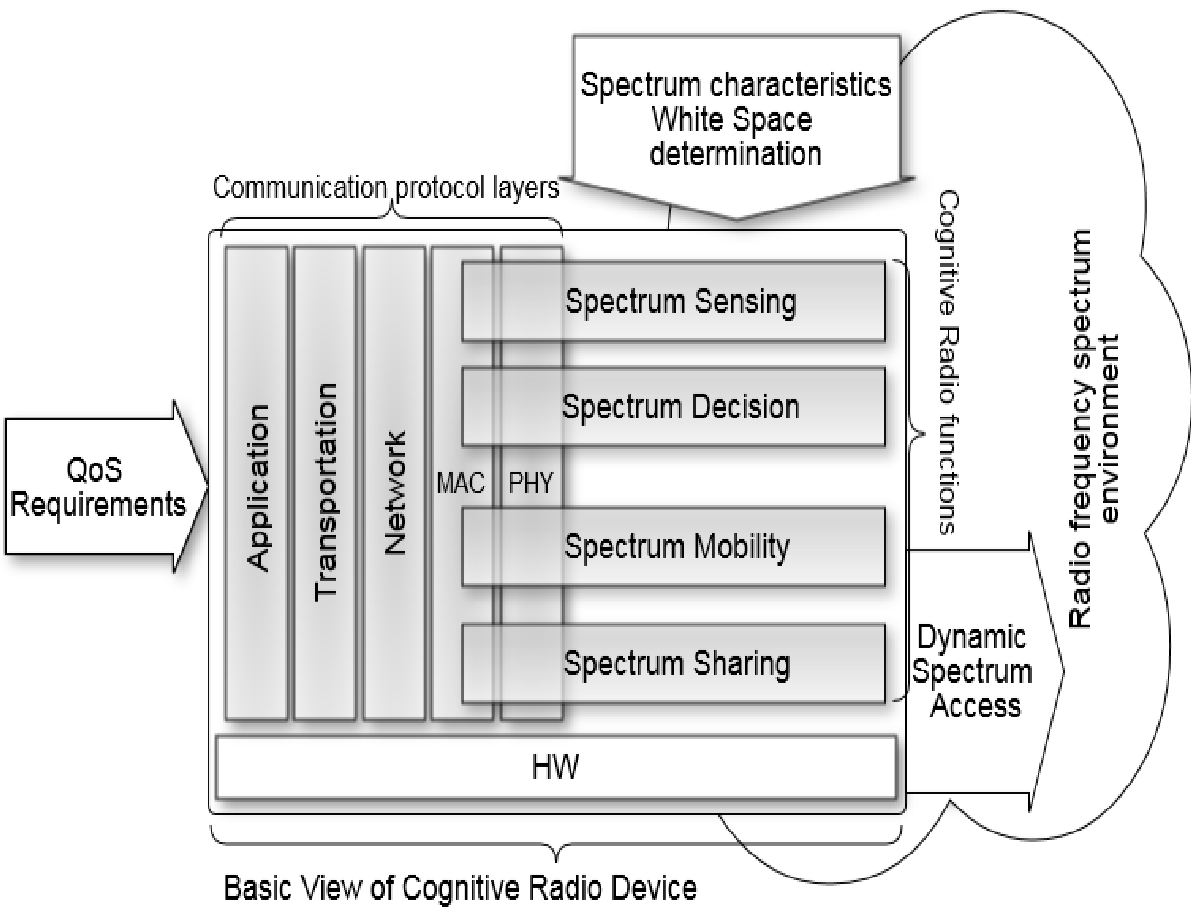


Figure 1.5 Basic CR device model

Security issues are another important area of research in CR, and recent surveys of these issues are provided in [57, 58]. Ultimately, the lower six OSI layers are designed to serve the seventh layer that provides the application or service through the underlying communication system. The success of any communication system is measured at the end by the level of QoS that can be achieved and how much this level satisfies the requirements of the provided service or application.

For successful CR applications, QoS requirements should be considered in the CR operations within an efficient mechanism for allocating the opportunistic resources [59]. Provisioning QoS requires the consideration of various factors and components, which makes it a complicated and challenging task. Specifically, the cross-layer concept of interacting between protocols at different OSI layers is needed in CR to attain the desired goals for RFS resource management and QoS provisioning [60]. CR performance is affected by PU activities that may not be predicted, which makes providing a QoS guarantee for SUs very challenging [59]. Providing solutions to address such CR challenges must take into account real-time operation requirements, system scalability, resource constraints and complexity [22].

1.2.6. CR standardisation efforts

At first, standardisation efforts focused mainly on using white spaces in the frequency bands regulated for television (TV) broadcasts. These bands have useful features that were thought promising in this early CR standardisation stage; some are listed below:

- There is high availability of white spaces in time and location, and in frequency domains particularly, since the switch from digital to analog TV transmissions.
- PU activities are usually based on scheduling operations which could be provided in advance to SUs through GDB.
- TV bands are located in low frequencies, less than 1 GHz. Such a range provides several favourable propagation characteristics that allow transmitted signals to travel longer and penetrate a much wider variety of objects than higher-frequency transmissions.

The basic TV channel widths are 6 to 8 MHz, and can be combined up to 32 MHz. The FCC has claimed that exploiting TV white spaces by CR technology allows free access for potentially more powerful public Internet connections and super Wi-Fi hotspots with greater coverage [61]. The first proposed worldwide CR standard was IEEE 802.22, released in 2011, to regulate MAC and physical layers specifications for wireless regional area networks (WRANs) operating in TV white spaces [62]. The standard is still being developed, and the last amendment was released recently, in 2018 during the writing of this thesis, to support enhanced broadband services and monitoring applications for operations in VHF/UHF TV bands, among other security and management enhancements, as reported in [63, 64]. The amendment also includes the ongoing effort to address the issue of different CR standards coexisting in the same white spaces. The second worldwide standard, IEEE 802.11af, released in 2013, was proposed to allow CR wireless local area networks (WLAN), also known as White-Fi, to operate in TV white spaces [65]. Both IEEE 802.22 and IEEE 802.11af were designed to use GDB as a mandatory approach to avoid interference with PU and identify available TV white spaces. The sensing approach is still optional in IEEE 802.22 for these purposes. In IEEE 802.11af, spectrum sensing is mainly used for sharing available channels with the competing IEEE 802.11-based devices, but it is still not clear how it may be utilised to identify PU transmission. The denominated CR standardisation effort, both nationally and internationally, is still limited to TV bands and based mostly on the GDB approach with a moderate consideration of the spectrum sensing approach, as reported in several surveys, such as in [60, 66, 67]. The efficient use of the spectrum based on CR becomes one of the key features for future wireless technologies. For example, the 3rd Generation Partnership Project (3GPP) standardisation group expresses interest in employing CR operations in future mobile networks like 5G LTE-advanced [60], and the IEEE 802.15.4m group is working on enabling low rate personal area networks to operate in TV white spaces [67]. This will allow more technologies to compete for the same white space, raising more challenges regarding coordinating coexistence and providing interoperability between systems. The Dynamic Spectrum Access Networks Standardization Committee (DySPAN-SC) [68], formally known as the IEEE Standardization Coordinating Committee 41 (SCC41), is currently working on addressing such challenges to improve spectrum use, and has proposed a set of standards and regulations under the IEEE 1900 initiative started in 2004 [66]. The next era is expected to move to other frequency bands rather than TV ones, as mentioned by FCC in [61]. Technically, moving to other bands will bring back the importance of spectrum sensing approaches, for implementing CR functions where the GDB is not easily provided.

1.3. Motivation

In the near future, the scarcity and underutilisation of RFS will become a severe problem, and improvement in CR research in terms of technologies, standards and business models to overcome the problem is essential. CR is considered a key technical component for the success of future wireless networks such as 5G [69]. In turn, the success of CR will be based on addressing the challenge in achieving the required levels of QoS to SUs while guaranteeing the necessary protection to PUs [59]. This study aims to contribute to this critical future technology. CR has to be able to operate in a wide range of wireless communication systems, such as cellular mobile and Wi-Fi networks, with different architectures. Typically, no single set of transmission techniques, such as signal modulation and coding schemes, transmission power and operating frequency, is suitable for all the available frequency bands and application requirements. A CR node should be designed to reasonably and intelligently select, and then reconfigure itself to operate with, the most suitable transmission techniques. In other words, a CR node can operate under different radio access technologies without the need for manually setting. In this study, this concept will be applied inside CR functions and communication protocols, specifically for spectrum sensing function. The sensing methods proposed so far have various advantages and drawbacks, and none can be claimed to be suitable for all CR operation scenarios. This study is based on the requirement that the CR system has to be capable of using diverse sensing techniques to provide a wide range of sensing capabilities. It should make dynamic use of the appropriate sensing method based on operating requirements and constraints, such as the various QoS requirements and CR device capabilities. Several of the research gaps that motivated this work are summarised below:

- The GDB approach is addressed in detail in current CR standards, while the spectrum sensing approach has not yet received sufficient consideration.
- Typically, the energy detection sensing technique is assumed when studying CR performance because of its simplicity and shortest sensing duration overhead.

However, its inability to distinguish PU signals from other signals is usually neglected while this limitation should be addressed.

- None of the proposed sensing techniques is suitable for all operational requirements.
- More studies are required to analyse QoS degradation in different applications and to find ways to reduce it when the sensing function is the only spectrum assignment approach under different
  - sensing methods represented in different sensing time and accuracy,
  - sensing strategies.
- The White-Fi node is not yet implemented in most of the popular simulation tools.
- QoS requirements should be considered before spectrum sensing.
- A mechanism for selecting the proper sensing method based on requirements and capabilities is required. The selection mechanism should be simple, effective and conducted in real time during CR operation.

This research is concentrated on White-Fi, or more generally CR networks based on IEEE 802.11 standards of random access, for many reasons. The IEEE 802.11af standard for White-Fi is the most recent worldwide CR standard, but it is still limited to TV bands, and the need for high-accuracy sensing to identify PU signals has not yet been addressed. The advantage of this standard is that the sensing is already used for random access under the well-known protocol, Carrier Sense Multiple Access/Collision Avoidance (CSMA/CA), to share the available free RFS bands (e.g., ISM bands), with other networks based on the same protocol. CSMA/CA will, therefore, be more suitable for CR devices to share available white spaces based on sensing. Wi-Fi networks are currently prevalent for use in Internet hotspots, so White-Fi networks are expected to replace Wi-Fi hotspots with super Internet hotspots. Even for future cellular mobile networks such as 5G, the trend is to exploit unlicensed bands, including the spectrum holes in licensed bands, by integrating with WLAN networks like Wi-Fi or White-Fi [70, 71]. The CSMA/CA is expected to be the preferred approach for implementing CR technology in different communication systems. Moreover, White-Fi technology is a promising approach to the solution of providing wireless communication to the coming Internet of Things (IoT) networks, last mile problems and mobile broadband connections. Providing solutions to White-Fi will have a high impact on addressing issues facing future networks and will contribute to their success.

1.4. Research questions and scope

The research questions are divided into one main question and three sub-questions.

The main research question: How can an efficient cross-layer mechanism for CR networks be designed to improve QoS for SUs while maintaining PU protection and achieving enhanced RFS utilisation?

The sub-questions:

First: Which CR function has the most significant impact on the QoS of the running applications?

Second: How can the spectrum sensing approach be conducted in such a way that it adapts to dynamic changes in QoS requirements and RFS conditions?

Third: How can the proposed solution integrate with MAC IEEE 802.11-based protocols for White-Fi to enhance QoS and improve spectrum sensing efficiency?

The main question and its first sub-question are addressed under the assumption that spectrum sensing is the only approach for assessing surrounded RFS. Therefore, any achieved improvement can be generalised to various spectrum bands and CR network architectures. The cross-layered design is limited to the correlation between the application layer and the CR sensing function implemented in the MAC and physical layers. This study has found that sensing is a major function, playing an important role in CR performance in general and QoS in particular. Accordingly, the second sub-question emerges and has been addressed under the concept of overlay spectrum sharing where a PU and SU cannot simultaneously share the same white space. Only local sensing techniques for detecting PU transmission, without the concept of cooperation between SUs in conducting sensing, are considered this work. The White-Fi network is chosen in this study to design a specific sensing strategy that is implemented in one of the existing communication techniques. Hence, the third sub-question aims at how to implement a solution for enhancing QoS while preserving a higher spectrum utilisation in this promising network. The sensing strategy is applied for both infrastructure and ad-hoc modes, based on the popular medium access mechanism, i.e., distributed coordination function (DCF), with the mandatory feature required in the current IEEE 802.11 standards, enhanced distributed channel access (EDCA). Addressing the research questions is conducted with the aim of minimising complexity without losing the scalability feature of the proposed solutions so that they can be implemented in a wide range of CR systems.

1.5. Research Contributions

The major findings and original contributions are listed below:

- The lack of standard classification of the possible outcomes and their accuracy of spectrum sensing in the CR context are addressed in this study. As a result, this study identifies three potential outcome levels of sensing techniques that should be considered in any CR studies involving sensing methods (see Section 2.3).
- The thesis points out the significance of using the spectrum sensing approach for assessing the surrounding RFS, and provides the cognition capability that will allow CR devices to operate in other frequency bands rather than being restricted to TV bands. In addition, there is a need to support more than one sensing technique in CR devices for better use of their features, as no existing technique is suitable for all CR operational situations.
- This work draws attention to the need for a dynamic sensing strategy able to adapt to real-time variant requirements of CR operations, in particular, QoS requirements (see Section 2.5).
- This work identifies the essential factors for designing an efficient sensing strategy for any CR system that attains the primary goals of employing CR technology (see Section 3.3).
- In this thesis, a CR node with different sensing strategies is implemented in a Riverbed Modeler (formerly known as Opnet) to study different performance metrics of the CR networks based on IEEE 802.11 standards (see Section 5.2).
- This work analyses QoS metrics such as delay and throughput in IEEE 802.11-based networks like White-Fi, when high accuracy is imposed under the denominated fixed sensing strategy; and shows that this conventional strategy is

not suitable to the dynamic nature of QoS requirements (see Sections 5.3, 5.4).

- This thesis proposes a fuzzy logic decision-making mechanism to select the proper sensing method in real time according to the defined essential requirements and constraints of CR networks (see Section 3.5). The solution can be implemented in many CR technologies because of its effectiveness and simplicity.
- This study verifies the settings and mechanisms used in White-Fi networks that should be considered for designing an effective spectrum sensing strategy (see Section 4.3).
- This work designs and proposes a novel spectrum sensing strategy, named the A QoS awareness MAC protocol (QACR-MAC), for enhancing the QoS in CR devices while utilising high-accuracy sensing methods, using the proposed fuzzy logic selection mechanism for better RFS utilisation and PU protection (see Section 4.4). QACR-MAC integrates with the IEEE 802.11e QoS mechanism used in IEEE 802.11 networks for improving the mechanism performance. The simulation results demonstrate a considerable improvement in QoS when the proposed sensing strategy, i.e., QACR-MAC, is used for a range of application requirements and network adjustments (see Section 5.5).

1.6. Thesis Stages and Outline

The research is conducted in four main stages, illustrated in Figure 1.6 (see next page), and these are briefly discussed in the following subsections. The last subsection outlines the remaining thesis parts.

1.6.1. Research gap investigation and knowledge base establishment

At this first stage, the information required for the research is collected from the literature and evaluated. The well-known sensing techniques for determining available spectrum holes are examined and classified, based on several metrics such as the information that they can provide about the spectrum, accuracy, operation time, and complexity. The investigation includes the QoS requirements for different applications: data, voice and video streaming.

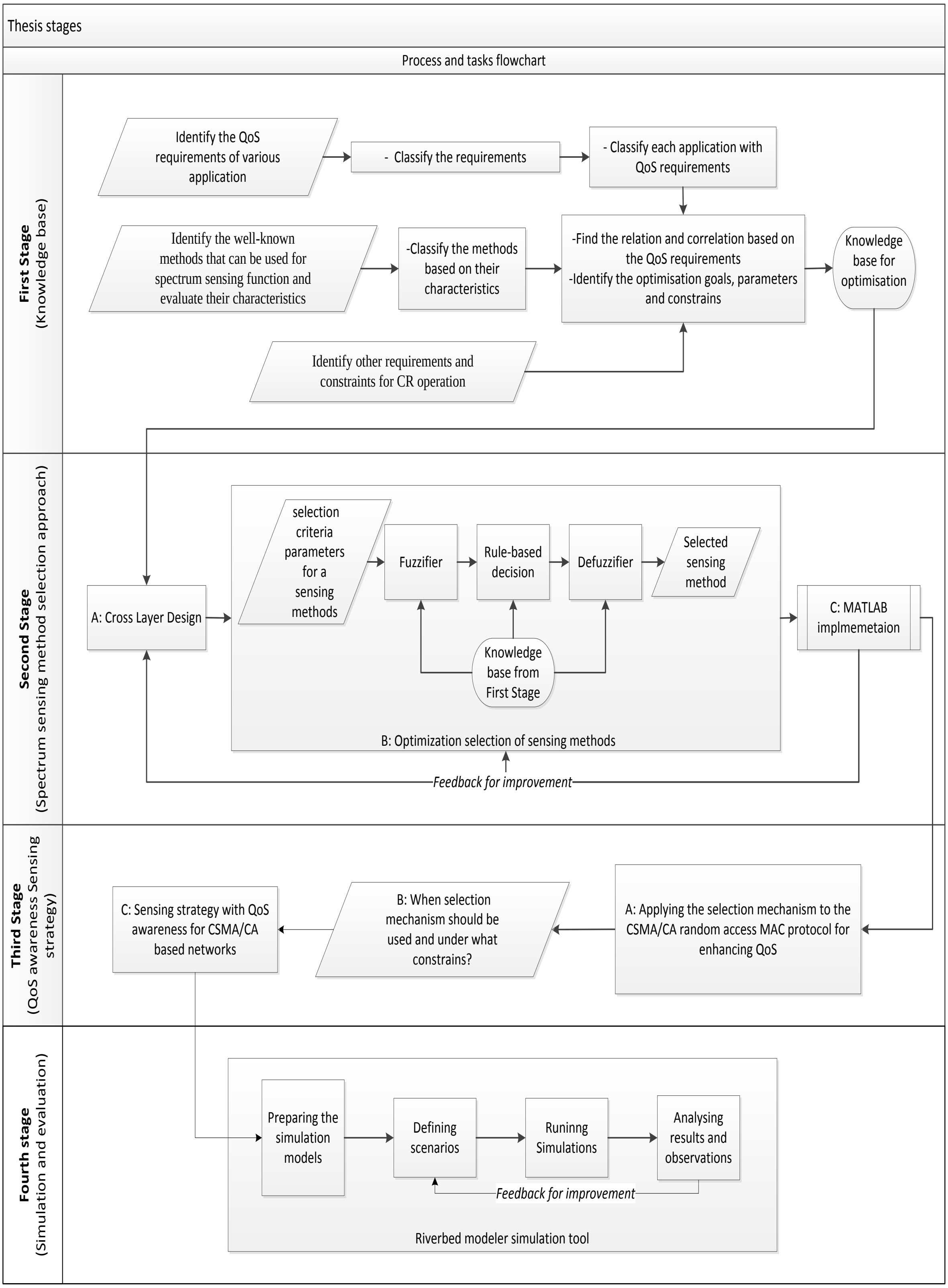


Figure 1.6 Thesis stages

The relevant QoS metrics, such as delay and throughput, are grouped and weighted for each application's set of requirements. The correlation between the QoS requirements and sensing parameters is also studied at this stage. The literature proposing strategies for conducting the sensing function in a variety of CR networks is examined, mainly for its discussion of sensing accuracy and QoS degradation. The results of this stage will provide the information needed to identify the required input and optimal output for each sensing function and QoS requirement, any possible constraints, and the parameters of CR node operations. Based on the results, identified relations and possible correlations will lead to the next stage, designing and implementing novel solutions.

1.6.2. Designing a sensing method selection mechanism

This second stage aims to design an effective mechanism by which a CR node can select the most suitable sensing method for the current operational situation. At this stage, a cross-layered concept will be used to consider the correlation between the sensing function at the MAC layer and the QoS requirements at the application layer (see Figure 1.6). Also, other factors that should be considered in the selection criteria of the mechanism are identified. The fuzzy logic algorithms are investigated, and then used to design the selection mechanism. This outcome of this stage is a new fuzzy logic decision-making mechanism to select the proper sensing method in real time during CR operations. The mechanism is implemented in MATLAB using the designer toolbox for designing a fuzzy inference system (FIS).

1.6.3. Proposing a QoS awareness sensing strategy for White-Fi

At this third stage, the resultant selection mechanism from the second stage is applied to CR MAC protocols based on CSMA/CA with EDCA for enhancing QoS. The issues of the current sensing method used in CSMA/CA are studied under the new requirements of CR operation, and those MAC operation parameters that might affect the sensing strategy are investigated. A sensing strategy (QACR-MAC) is proposed to minimise the overheads from using high-accuracy sensing methods by using the proposed selection mechanism that participates positively in the used performance MAC protocol settings. QACR-MAC is designed to integrate with the IEEE 802.11e mechanism to enhance QoS in CR IEEE 802.11-based wireless networks.

1.6.4. *Simulation and evaluation*

At this fourth stage, the Riverbed Modeler (version 18.0.1) is used to implement the CR node with various sensing strategies to evaluate the QoS metrics of different applications under a range of diverse sensing parameters. The proposed solutions from the second and third stages are evaluated under different MAC protocol settings and network designs. The simulation results of this stage are checked to see if they validate the outcome from the previous stages. Several simulation scenarios are designed to evaluate the performance of the proposed solutions of this thesis regarding enhancing QoS under different operational settings. In particular, the simulations are used to analyse the QACR-MAC performance regarding enhancing 802.11e, different frame aggregation configurations, coexistence with other SU systems, and scalability.

1.6.5. *Thesis outline*

The remaining parts of this thesis are organised as follows:

Chapter 2 presents the research conducted in the first stage of the literature review and the base knowledge foundations of this work that are particularly related to spectrum sensing and QoS.

Chapter 3 describes the fuzzy logic decision-making mechanism proposed in the second stage to select an appropriate sensing method.

Chapter 4 discusses sensing issues in White-Fi networks and describes the QoS awareness sensing strategy, QACR-MAC, proposed in the third stage for improving QoS.

Chapter 5 demonstrates the simulation work of the last stage and discusses the key findings and results of evaluating the proposed solutions.

Chapter 6 presents this thesis summary and conclusions and potential future work.

Appendix A documents the FIS design and implementation conducted in MATLAB.

Appendix B documents the implementations and some of the simulation scenarios configurations carried out in the Riverbed Modeler.

Chapter 2. Sensing function and QoS

The sensing function is the most important function in the Cognitive Radio (CR) operation cycle, as discussed in Section 1.1. All other CR functions are based on its outcome. This function has to be conducted periodically during CR operation to recognise the surrounding Radio Frequency Spectrum (RFS). This chapter aims to examine the main features and limitations of available sensing techniques and strategies used for CR that appear in the literature, especially features and limitations related to the Quality of Service (QoS). Another aim of this chapter is to identify research gaps, which will lead to the development of novel solutions proposed and described in the following chapters. This chapter begins with a brief introduction of the need for effective spectrum sensing, in Section 2.1. Several of the well-known spectrum sensing techniques are classified and defined in Section 2.2. These are compared by characteristics and performance in Section 2.3. The impact of the sensing parameters on QoS is discussed in Section 2.4. The proposed spectrum sensing strategies for CR in the literature are identified and evaluated in Section 2.5, and a summary of the chapter is provided and discussed in Section 2.6.

2.1. Introduction

From a technical perspective, the CR concept is a promising technology to achieve efficient utilisation of RFS. Detecting the presence of a Primary User (PU), or more precisely finding out whether the PU is using its allocated spectrum or not, is an essential task for a CR device. The sensing function has to be conducted periodically during CR operations to comply with RFS regulations. On one hand, this fundamental task requires improved sensing accuracy by avoiding false positive results while detecting the presence of a PU. On the other hand, the sensing technique should achieve a high probability of detection of available spectrum holes. The nature of electromagnetic signals makes accurate sensing a complicated process. More specifically, the signal-to-noise ratio (SNR), the multipath fading of the PU signals, and the changing levels of noise can significantly affect sensing accuracy [28, 29]. Moreover, the cognition capability and the consequent actions are based on the level of detail and accuracy of the information gathered about the RFS. Therefore, imperfect spectrum sensing can result in increased transmission error rates, for both the PU and the secondary user (SU) [37], which may contribute to the degradation of the QoS provided by a

PU and SUs. Any QoS degradation that can be attributed to the CR technology can potentially harm the progress of CR-based solutions, so studying the performance of available sensing techniques and strategies with an eye to QoS is an essential step in proposing an effective solution for CR solutions.

2.2. Spectrum sensing methods

Spectrum sensing can be based on different methods. The sensing technique is a combination of a hardware portion of a CR device used to collect information from the surrounding RFS, and an algorithm used to analyse the collected information. The basic aim of simple sensing techniques is to detect if a channel is idle or not. For advanced detection, for instance to distinguish a PU signal among other signals, more complex techniques are needed. A sensing method may involve more than one sensing technique. In the literature, the term 'local sensing method' is used to refer to the technique used to gather RFS information around a single device. Sensing techniques that do not require any prior information for executing their algorithms are known as blind sensing; two of these well-known techniques are explained in Section 2.2.1. Other, non-blind sensing techniques that require prior information about the detected signal or noise variance for their operations are described in Section 2.2.2. Some of the non-blind sensing techniques could be considered semi-blind sensing when the required prior information is minimised. When the sensing function outcome is based on collecting RFS information from several CR nodes, it is usually called a cooperative sensing method; this is discussed in Section 2.2.3.

2.2.1. Blind sensing

In blind sensing, no prior information about a PU signal is necessary, although prior information about the noise power of the targeted spectrum may be required for better performance. Otherwise, a reasonable estimation of the noise power is sufficient. Two well-known blind sensing methods are based on energy and covariance-based detections.

2.2.1.1. Energy detection

Energy Detection (ED), also known as radiometry or periodogram, is the most common method for spectrum sensing because of its low implementation complexity and computational overhead [28]. In this method, an energy detector is used to monitor the energy level of the communication channel with a narrowband frequency spectrum and then the observed signal energy level is compared with a predefined threshold. The channel is busy if the signal's energy is over the threshold. Otherwise, it is considered idle. When this technique is used for CR, the busy state is assumed because of the PU transmission, which may not be the case as the busyness could be from other SUs' transmissions. Because of its simplicity, this technique requires the shortest sensing time compared to other common sensing technologies [72]. Generalising the use of this method in CR faces several challenges as a consequence of its simplicity. First, selection of the threshold used for detection becomes an issue when the channel noise level is unknown or uncertain over time [73]. Second, under a low SNR, it is hard to differentiate between modulated signals, including signals of other SUs, noise, and interference, resulting in poor detection performance [28]. Lastly, an energy detector is ineffective in detecting spread spectrum signals [74].

2.2.1.2. Covariance-based detection

This method is based on comparing the covariance of the observed signal and the covariance of the noise where statistical covariance matrices of signal and noise are usually different [75]. The main improvement of this method is to overcome some shortcomings in energy detection (ED). In particular, it can distinguish between signal and noise in a low SNR to some extent, and without any prior information about the PU's signal and channel noise. This improvement is achieved at the expense of a computational overhead in computing the covariance matrix of the observed signal samples [76]. In addition to an increase in complexity, other drawbacks of ED are present in covariance-based detection, such as the inability to distinguish between different transmitted signals.

2.2.2. Based on prior information

Methods belonging to this category rely on partial or full information about the PU's transmission signal to be able to differentiate it from other signals and noise. In this category, the sensing method takes several samples of the surrounding spectrum of a certain channel and extracts any possible features of the collected samples. Then it compares these characteristics with the prior known features of the PU signal. The algorithms used to derive and compare these features differ from one sensing method to another. The amount and type of the required prior information depend on the algorithm used. In this section, several of the sensing methods that belong to this category are described.

2.2.2.1. Cyclostationary feature detection

This method is based on distinguishing a signal from noise and interference by identifying its cyclostationary features associated with the signal modulation type, carrier frequency and data rate [77, 78]. The CR device needs sufficient prior information about these unique characteristics of the PU signal, and using this information it can perform a cyclostationary analysis on the signal to identify matched features [79]. Some research considers this technique blind or semi-blind because it requires only some fundamental information, which could be predicted, about the signal [80, 81]. However, this may only be valid when the aim of using this technique is to distinguish a signal from noise, but when used to distinguish between different signals, higher prior information is required. For this method to perform better than ED, an adequate number of real-time sample sets in the frequency domain need to be collected [82]. The improvement effort for this detection technique is on reducing the amount of required prior information and the computational complexity of its algorithm.

2.2.2.2. Correlation detection

Sensing based on correlation is also known as waveform-based sensing or coherent sensing. In this method, the expected correlation or coherence between observed signal samples is identified, to then identify the PU signal based on previous knowledge about its waveform patterns [30]. The accuracy of the sensing increases when the length of the known signal pattern of the PU is increased [83]. The main drawback to this method is related to the large amount of information required about the PU signal patterns to achieve a high performance [84]. As this level of information may not be available in all CR systems, this method is not practical for all CR scenarios.

2.2.2.3. Radio identification based sensing

This method is based on having a priori information about the transmission technologies used by the PU. In the radio identification stage of this method, several extracted features of the received signal are identified and then classified to indicate whether if the signal demonstrates the PU signal or other signals [85]. Such feature extraction and classification techniques are used in the context of the European Transparent Ubiquitous Terminal (TRUST) project [86]. To collect and extract the signal features, the radio identification method may use known sensing techniques, such as ED [30]. The radio identification improves the accuracy of ED to some extent. The degree of precision is dependent on the techniques used to extract, classify and identify the PU.

2.2.2.4. Matched filter detection

The matched filter method achieves a higher detection probability in a shorter detection time than other methods similarly based on prior information [87, 88], and is considered the best sensing method in this classification. The collected signal is passed through a filter that amplifies any PU signal and attenuates any noise signal, making detection of the presence of the PU signal more accurate [88]. The filter, known as a matched filter, has to be tuned according to some features of the PU signal such as the required bandwidth, operating frequency, modulation used and frame format [30]. One of the disadvantages of this method is in implementation, where different PU signal types require different dedicated hardware receivers. This makes the method impractical to implement and leads to higher power consumption during operation if it is based on current hardware technologies.

2.2.3. Based on SU cooperation

Another spectrum sensing approach is based on cooperation between SUs. The main principle of this approach is that SUs share their locally sensed information about the spectrum with each other. Using information from other SUs can produce a more accurate sensing outcome than relying solely on local sensing. The hidden terminal problem is an example of an issue that may prohibit an SU node from detecting the presence of a PU when the CR network relies only on locally sensed information. The cause of this problem is the fading and shadowing of signals emitted bby a PU, even when they are within the SU's transmission range [30]. When cooperating SUs are spatially distributed, it helps to overcome the hidden PU problem and other limitations of local sensing [89-91]. Sensing cooperation can also reduce local sensing costs, e.g., by sensing time and energy consumption while maintaining sensing quality by scheduling the sensing operation among cooperative SUs [92]. The sensing method used by an individual SU can be based on one of the sensing techniques for local sensing, such as ED and cyclostationary feature detection [93].

In some environments, cooperative sensing may lose its advantages as far as an individual SU is concerned. For instance, increasing the local sensing frequency in an individual SU of high mobility is more efficient, regarding sensing accuracy and overhead, than to cooperate with other SUs [74]. In cooperative sensing, the improvement of sensing is more noticeable when the number of cooperative SUs is increased. However, the involvement of more SUs will increase the cooperation overhead in terms of the amount of data exchange and the time required for the exchange [94]. Moreover, the cooperative approach can only be used when SUs are able and willing to collaborate. Practically, an SU may not always find other cooperative SUs within its transmission range in distributed wireless networks, so CR devices should not rely solely on cooperative sensing approaches. As a consequence, the cooperative approach is out of this study's scope.

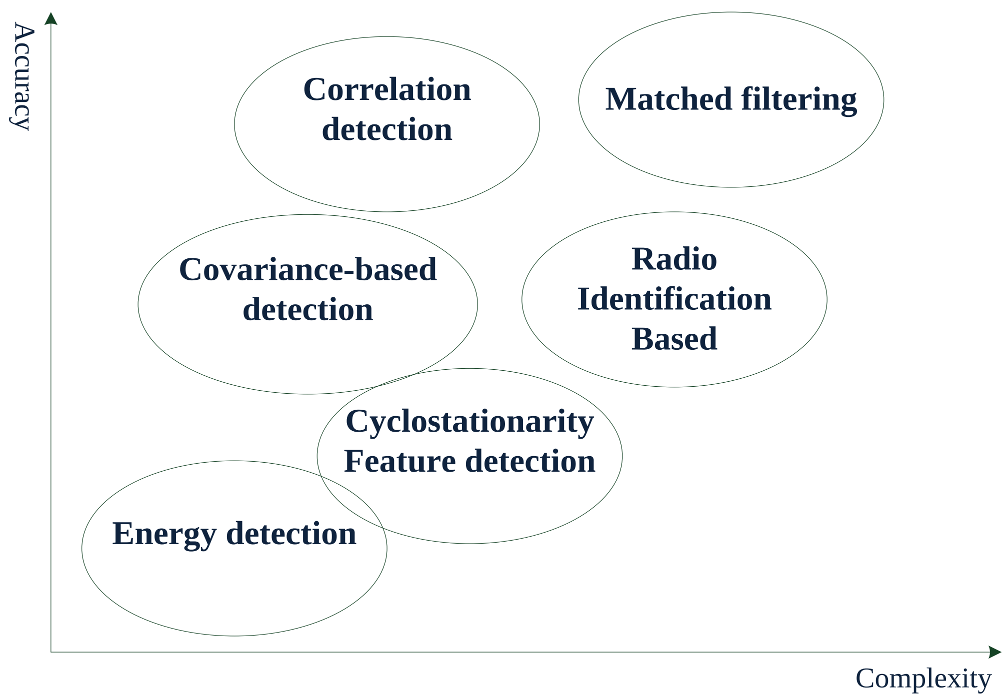


Figure 2.1 Sensing methods complexity versus accuracy

2.3. Performance comparison of spectrum sensing techniques

Several criteria could be considered when comparing different sensing methods. In this section, the most important and relevant characteristics of the methods related to this study are discussed. Accuracy and complexity are the most common essential factors used to compare sensing methods. Figure 2.1 (above) shows a stylised comparison between local sensing techniques, based on the accuracy and complexity of the sensing methods reviewed in this study and derived from several sources, including [30], [95] and [96].

Detection accuracy is usually represented by two factors: detection probability (P_d) and false alarm probability (P_f). P_d is the probability of correctly detecting the presence of a signal or more in the channel being sensed. Thus, the probability of missed detection (P_m) is (1- P_d) where the detection method fails to detect the presence of a signal or signals. P_f is the probability of detecting a signal as present when it is, in reality, absent. The accuracy of a sensing technique is based on the achieved P_d and reduced level of P_f.

The nature of electromagnetic signals makes accurate sensing a complicated operation. Specifically, the uncertainty of noise level and signal-to-noise ratio (SNR), and the multipath fading of PU signals, can significantly affect sensing accuracy [29]. The SNR represents the ratio of the power of a signal to the average noise power at a given point in a channel, and is usually measured in decibels (dB). The SNR of a channel has a direct impact on the P_d and P_f of the sensing technique used; and therefore the accuracy of a sensing method depends on the SNR. Under a given SNR, each sensing technique has a mathematical correlation between its P_d and P_f values. Usually the correlation between P_d and P_f is represented by curves in logarithmic scales called the receiver operating characteristic (ROC) curves. MATLAB, the well-known technical programming and simulation tool, provides a set of functions to study matched filter detection including its ROC [97]. By using MATLAB, ROC curves, shown in Figure 2.2, are generated to illustrate the correlation between P_d and P_f for matched filter detection. The ROC curves represent different SNR values: 10dB, 5dB, -15dB and -20dB, when the channel noise is modelled as additive white Gaussian noise. Figure 2.3 shows the ROC curves for ED under -20dB and -15dB SNR values when the channel noise is modelled as additive white Gaussian noise based on the MATLAB code provided in [98].

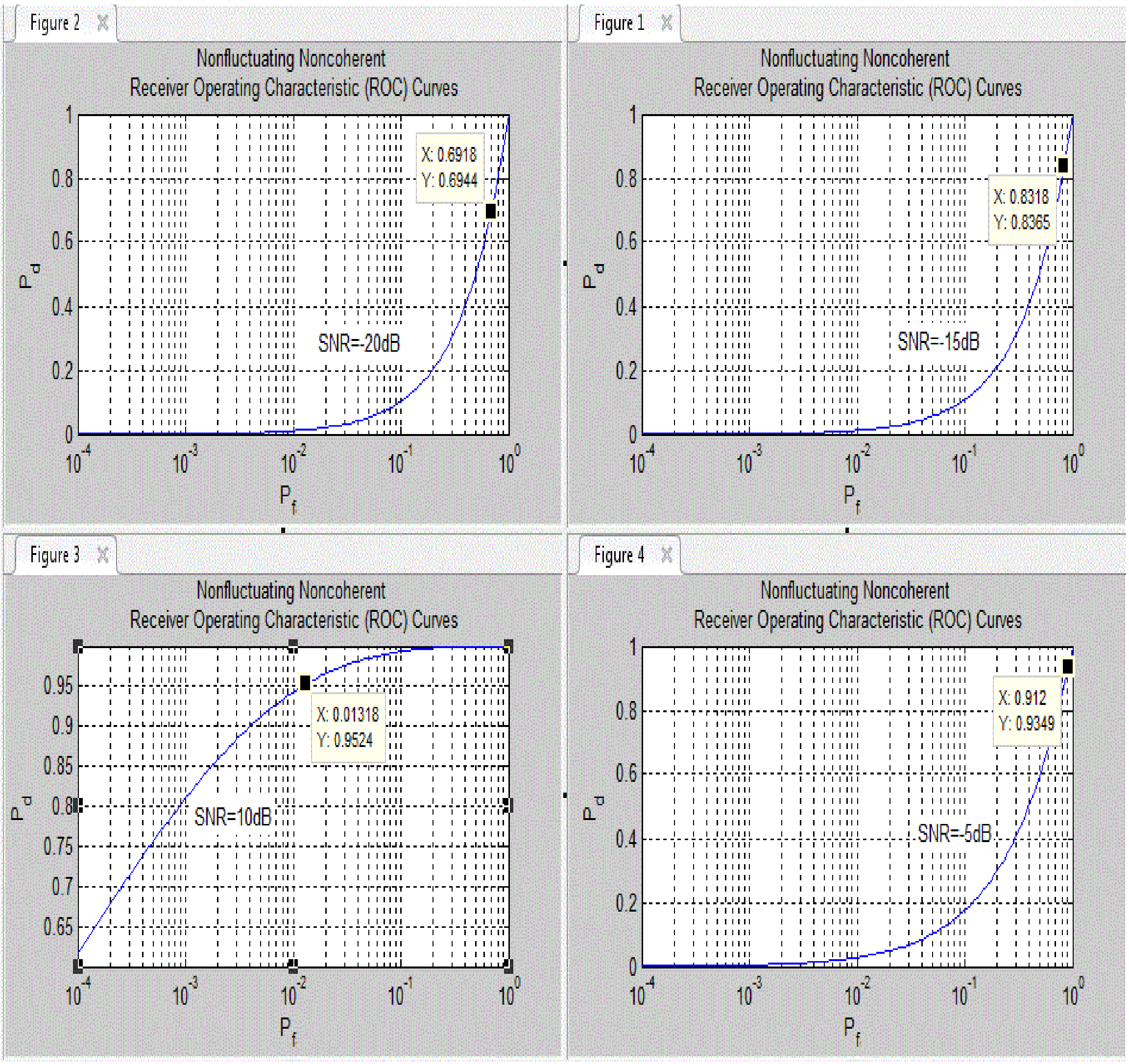


Figure 2.2 The ROC for matched filter sensing under different SNR values under a white Gaussian noise channel (the graphs generated in MATLAB by using built-in code)

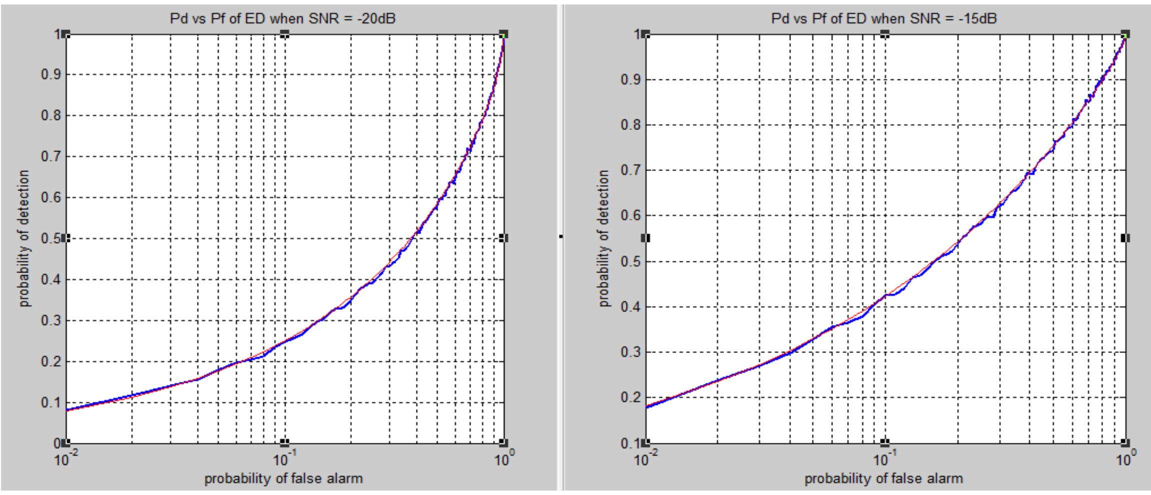


Figure 2.3 The ROC for ED sensing under -20dB & -15dB SNR values under a white Gaussian noise channel (the graphs generated in MATLAB based on the code proposed in [98])

The ROC curves show that improving P_d results in increasing P_f , particularly at a low SNR. In general, P_d is increased and P_f is reduced as the SNR increases for a given channel. For the same SNR, different sensing techniques may achieve different accuracy; i.e., P_d and P_f, under the same channel conditions. According to the IEEE regulations for sensing in TV white space, P_m and P_f should be less than 0.1 and 0.9 respectively [99]. Complying with such regulations may not be possible under a certain minimum SNR, called critical SNR. Sensing methods have different critical SNR values. For example, it has been found that in a given channel condition, the critical SNR values for ED and matched filter detection are -16dB and -30dB respectively, for the detection probability P_d equal or larger than 0.9 [100]. Sensing accuracy could be improved under low SNR by increasing the sensing duration.

The **sensing duration** or sensing time is another important performance factor. A minimum sensing duration is required for conducting a spectrum assessment via a sensing technique. For instance, ED requires the shortest duration of the better-known sensing techniques [72]. More advanced techniques require much longer to achieve acceptable accuracy. In general, increasing sensing duration helps achieve higher accuracy under the same channel conditions. However, increasing sensing duration is constrained by several factors, and an optimal time is required. In Section 2.2.3, the QoS constraint for sensing duration is discussed in more detail. Increasing duration may not help to achieve acceptable detection levels under a certain SNR threshold known as the SNR wall [101]. This wall differs from one sensing technique to another, based on the robustness of their algorithms against noise level uncertainty. ED is claimed to have the highest—that is, the worst—SNR wall, while sensing techniques based on identifying the observed signal feature, such as cyclostationary feature detection, can achieve a lower—that is, better—SNR wall [102]. The number of channels that can be sensed to identify possible spectrum holes will depend on the sensing duration. Increasing the sensing duration will help to sense more channels. Diverse sensing techniques usually need different lengths of time to sense the same number of channels, as the minimum required time for each channel may vary. For instance, the maximum number of sensed channels during two seconds of sensing period was found to be 2,602 for ED and 72 for matched filter under conditions reported in [100].

The **distance** between the source of the transmitted signal or signals and the node of sensing is a fundamental factor affecting sensing accuracy. The strength of the electromagnetic signals used in the RFS attenuates exponentially with the propagation distance. Proportionately, the SNR of a signal decreases with the signal propagation distance because of the aforementioned attenuation phenomenon in radio signals. As a consequence, detection accuracy depends on how far the sensing node is from the transmission source, and which sensing method is used. The typical SNR of a radio signal and the relative sensing behaviour is illustrated in Figure 2.4. Most of the sensing methods perform well; i.e., $P_d > 0.9$ and $P_f < 0.1$, when the SNR is high, in terms of distinguishing signals from noise. However, not all sensing methods can distinguish between different signal waveforms and modulations [27], and those techniques with such a limitation cannot distinguish PU signals from SU signals. For example, blind sensing methods cannot effectively distinguish between a PU and SUs that are sharing the same spectrum holes. For advanced sensing outcomes, more complex analyses of the signals collected have to be undertaken and compared with prior information about the PU's transmission signal.

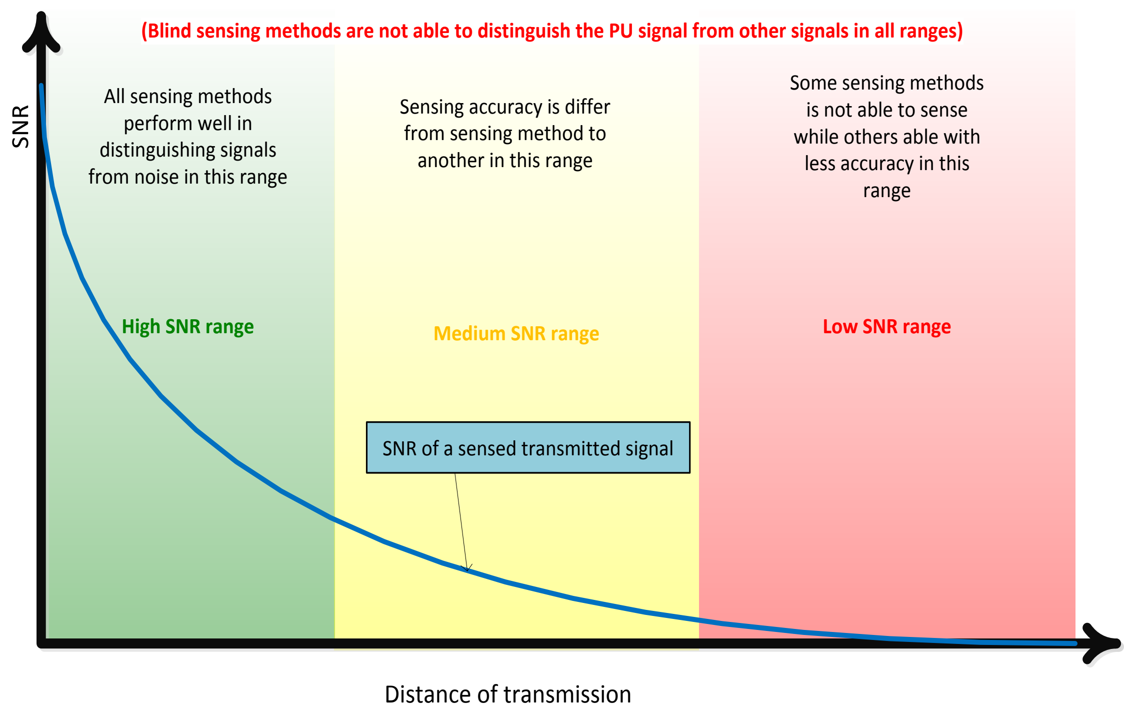


Figure 2.4 Typical SNR vs distance of a radio signal and the relative sensing behaviour

Sensing outcome is another performance factor that needs more consideration when comparing sensing methods. The core concept of CR is the ability to recognise the RFS conditions and adapt transmission accordingly. The cognition in CR depends on how many details about the surrounding RFS the sensing method can provide. In other words, the level of cognition gained about the surrounding RFS depends on the details and accuracy of the sensing outcome. This outcome can be categorised into three levels based on the detection capabilities, as summarised in Table 2.1. The first level is based on detecting whether the channel is occupied by any signal or is idle [28]. The sensing outcome, in this case, may only decide if the channel is busy or idle. The given P_d and P_f for this basic task of detection are representing, respectively, the probability successfully detecting the busyness of the channel and the probability of a false detection. Hence, the P_d and P_f are not related to detecting PU signals in specific, as is widely assumed in CR research. For instance, the P_d and P_f, shown in Figure 2.3, for ED are about the accuracy of detecting if the channel is busy, regardless if it is because of PU or other signals, or idle.

The second level of outcome is the ability to decide if the channel is occupied by the PU or not. The third level of outcome helps to distinguish between signals present in the channel, whether of a PU, an SU, or an interfering signal/noise [27]. The second and third levels are based on the available information about the PU signals and other potential SU signals as well as the used sensing technique and hardware technology. Sensing methods that are based on prior information are better able to differentiate PU signals from other signals and noise. Among the well-known methods, the matched filter is considered the best method regarding accuracy, as it achieves a higher detection probability of PU signals [87, 88, 103]. A comparison summary of the reviewed sensing methods in this study, based on their characteristics presented in the literature, is illustrated in Table 2.2. The comparison shows that each feature gained by a sensing method comes with the loss of another feature or more. For instance, higher accuracy causes more complexity, longer sensing duration and more required information.

Table 2.1 Possible sensing outcome levels

Sensing outcome	**First Level (Basic)**	**Second Level**	**Third Level**
Capabilities	>To decide if the channel is busy or idle. >To distinguish between transmitted signals and noise. >The accuracy of this level depends on the extent of the ability to distinguish signals from noise.	>To distinguish PU signals from other signals in the channel. >The number of PU signals that can be detected is based on the information provided about each different PU signal and the accuracy of the sensing technique.	>To distinguish between PU signals and other SU signals in the detected channel. >The scope of this level is based on the available information about the presented signals and the accuracy of the sensing technique.
Examples	Blind sensing methods	Correlation detection with prior information about the PU signal	Matched filter with prior information about all detected signals and the required hardware.

Table 2.2 Comparison between the reviewed local sensing techniques

Sensing method	Detection accuracy	Prior information required	Sensing duration	Robustness against SNR	Complexity
Matched filter	**High**	High	High	High	**High**
Correlation	**Medium**	Medium	High	High	**Medium**
Radio identification based	**Medium**	Medium	Medium	Medium	**High**
Cyclostationary feature detection	**Low**	High	High	High	**Medium**
Covariance	**Low**	Low/none	Medium	Medium	**Medium**
Energy detection	**Low**	Low/none	Low	Low	**Low**

2.4. Impact of sensing parameters on QoS

CR technology can be used to support applications that generate different types of traffic, such as data, voice and video. Table 2.3 shows potential applications of CR and their sensitivity to some of the QoS metrics, based on related studies in [104-106]. As there are no widely accepted standard measurements of QoS requirements for possible applications in CR, Table 2.3 is used in this thesis as such a standard.

Table 2.3 Potential applications in CR and their sensitivity to some of the QoS metrics

Application Type	**Examples**	**Sensitive to**			
		Delay	Jitter	Packet loss	Throughput
Real time	Video conferencing	High	High	Low	High
	Voice conversation	High	High	Low	Low
	Video streaming	Low	N/A	Med	High
Non-real time	E-mail	Low	N/A	High	Med
	File Transfer Protocol (FTP)	Low	N/A	High	High
	Internet browsing	Low	N/A	High	High
Message-based	Internet relay chat	Med	N/A	High	Low

The eighteen parameters that have the most impact on the QoS in a CR network are identified in [14] as :

"1) Transmission Power
2) Bandwidth
3) Delay
4) Delay jitter
5) Throughput

6) Mobility

7) Handoff

8) Bit error rate (BER)

9) Signal strength

10) Signal to interference and noise ratio (SINR)

11) Dynamic availability of idle channels

12) Expected Holding time of idle channel

13) Spectrum efficiency (bit/s/Hz)

14) Interference mitigation

15) Degree of complexity

16) Amount of overhead (cost)

17) Reliability

18) Low power capabilities of mobile devices.” [14]

Most of the above parameters are influenced directly or indirectly by the sensing operation. The QoS parameters that are directly affected are delay, jitter and throughput, mentioned in Table 2.3. Typically, a CR device cannot transmit on a channel while sensing it, first because a CR device senses a channel if it is vacant, and second because it is impractical for a CR to distinguish between its own transmissions and those of a PU. Generally, sensing is conducted periodically during a CR operation to avoid interference with the PU and to identify spectrum holes. The transmission of data starts from the end of one sensing time until the start of the next. One period, consisting of one sensing period and one data transmission, is called a CR frame. A simple structure of the CR frame resulting from this periodic sensing is shown in Figure 2.5.

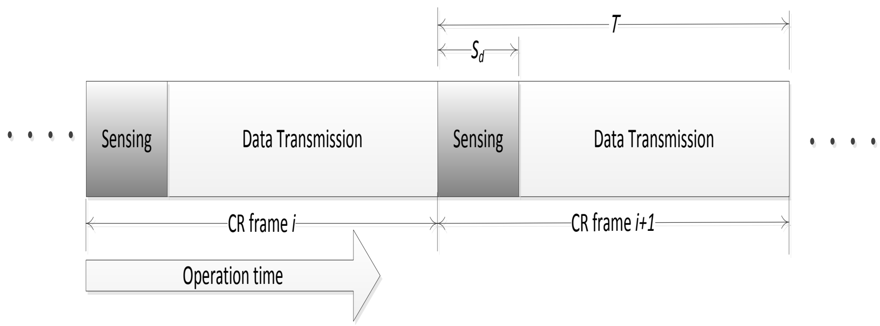


Figure 2.5 Simple structure of CR frames based on sensing operation

The CR frame duration *T* represents when the sensing will be re-conducted; thus sensing frequency is *1/T*. *T* also represents the communication frame length in CR. The transmission time (T-S_d) depends on the sensing duration S_d and the frame duration *T,* based on how frequently the sensing is conducted. The sensing duration S_d and the frame time *T* can be designed to be fixed for all frames or can be designed to vary depending on the design goals [107]. After each sensing, the CR device can decide to transmit data on the same channel or to use a new vacant channel, in response to the sensing outcome; i.e., the presence or absence of the PU. Increasing the sensing duration S_d and conducting the sensing more frequently; i.e., decreasing *T*, will improve the probability of correct detection of the PU presence. This leads to more protection of the PU's signals from interference by SU transmissions, but also to QoS degradation for CR users. This degradation can be measured by several parameters such as throughput, delay and MAC layer process overhead [108]. The sensing operation should be as short and as infrequent as possible without affecting its accuracy, to maximise the time for data transmission [32]. Designing the time and frequency of sensing should, therefore, take into account the trade-off between protecting the PU's QoS and improving the QoS of CR users.

The level of protection required for the PU's QoS may vary depending on the available frequency bands and types of service. For instance, an analog TV service is more robustness against interference than a digital service [109]. Some of the QoS challenges and requirements for SUs and PUs are pointed out in [59] and listed here in Table 2.4. Most are associated with sensing operation and outcomes. As the sensing function plays an important role in the QoS requirements for both SUs and PUs, it is essential to investigate the impact of the sensing operation on the QoS and find the optimum way to reduce it. In general, for optimal sensing performance, increasing the detection probability is constrained by the acceptable level of false alarm probability and vice versa when designing a sensing strategy [110]. Additionally, improving sensing performance is challenged by a range of trade-offs between sensing accuracy and a variety of constraints such as application requirements, hardware capability, complexity, and the infrastructure used [109].

Table 2.4 QoS challenges and requirements for an SU and a PU [59]

QoS requirement for a SU	QoS protection for a PU
The required level of QoS should be maintained in the presence of variations in the available spectrum resources.	CR networks shall have the capability to evict users upon resource shortage, for example, due to PU appearance.
The quality of radio environment awareness shall depend on the QoS requirements.	CR networks shall schedule quiet periods for spectrum sensing purposes, when required, without degrading the QoS of the SUs.
The reserve channels to be used when a PU is detected shall be identified according to QoS needs.	CR networks shall vacate their operating channel upon the appearance of a PU, or use a transmit power allowing simultaneous operations (underlay sharing).
The admission control shall depend on the QoS requirements and on the channel capacity of the available resources.	The transmission of information related to the presence of a PU shall have priority over data transmission of SUs.

Regarding QoS constraints, available studies are still not enough to address this issue in different CR systems and provide a standardised QoS mechanism. The key sensing parameters that impact on QoS are the sensing duration, outcome level (see Table 2.1) and accuracy. These parameters and their performance vary depending on which sensing method is used. However, most available studies of the impact of one or more sensing parameters on QoS assume the methods use ED techniques. The sensing duration and its impact on the achievable throughput, when ED is assumed for sensing, were studied in [110-115]. The maximum sensing time considered in these studies was up to 100 ms to find the optimal duration to achieve the highest throughput under specific conditions. For instance, the optimal sensing duration for the IEEE 802.22 network was found to be 14.2 ms in the bandwidth channel of 6 MHz with SNR = -20 dB when the CR frame duration is 100 ms and P_d = 0.9 [110]. The impact of the sensing parameters on the delay of potential applications running on CR nodes has not received the same attention from researchers as for throughput. Several studies, also based on ED, analysing the delay experienced in SUs were reported in [116-119]. The main concerns about the results of the studies mentioned above, and any studies based on ED, are the limited abilities of the sensing method and its accuracy in low SNR. The given accuracy P_d for ED does not reflect the accuracy of detecting the PU transmission. In fact, it only reflects the accuracy of detecting the busyness of the channel—which could be because of the other SU transmissions and not because of the PU's appearance. In such as case, the response action may be inappropriate. For example, an SU is not required to leave a channel if it is busy only because of other SU transmissions.

The need to use a more complex sensing method to overcome ED limitations has been pointed out and considered in some research, such as in [102, 120, 121], to achieve improved sensing outcomes. However, using more accurate sensing methods requires a longer sensing time, in addition to the complexity overhead. Such consequences of improving sensing outcome and increasing its duration on QoS should be considered in finding the most effective sensing method. For instance, the relation between sensing accuracy and sensing time or frequency was the primary focus of the authors of [122], who suggested that most attention should be on finding an optimal spectrum sensing technique with the capability of flexible tuning between time and frequency resolutions. For such possible optimisation, they nominated a wavelet-based spectrum estimation method. However, they found that available state-of-art sensing technologies do not offer a possible trade-off between the complexity of a sensing method and its accuracy. Other researchers studied other sensing techniques for improving sensing accuracy for CR, such as correlation-based sensing in [84], cyclostationary features in [123], and matched filters in [124]. Most of the effort of using advanced methods was on improving the sensing accuracy and overcoming ED limitations, without considering QoS constraints. In general, the dominant approach has been to find an optimal sensing method and readjust its parameters for all possible CR operations. However, all the researchers noted above agree that none of the available sensing methods is suitable for all possible sensing situations, conditions, and technologies of CR systems. Employing sensing methods in different CR technologies implies a need for various sensing strategies; and these are discussed next.

2.5. Spectrum sensing strategies

The design of the sensing strategy aims to conduct spectrum sensing appropriately by determining when and for how long it should operate, and what it should sense, in addition to which sensing method to use. The strategy should be designed according to the Media Access Control (MAC) protocol in use. In general, MAC protocols take one of three approaches: random access, time-slotted and hybrid-based [52, 125]. Each approach can be established in centralised or distributed mode. In centralised mode, the wireless nodes communicate through a base station. In distributed mode, also called ad-hoc mode, they communicate with each other without a base station. Examples of the proposed CR MAC protocols for each approach are listed in Table 2.5.

The carrier sense multiple access with the collision avoidance mechanism (CSMA/CA) is widely used in random access and hybrid-based protocols. The well-known IEEE 802.11 standards including the CR one, IEEE 802.11af, are based on CSMA/CA. In IEEE 802.11, the distributed coordination function (DCF) protocol is used for random access while the point coordination function (PCF) protocol is used for the hybrid-based approach. When a CR device uses the CSMA/CA to share the same spectrum holes, the sensing operation will be conducted more often.

Table 2.5 CR MAC protocol classification and examples.

Network mode	CR MAC protocol approach		
	Random Access	**Time-slotted**	**Hybrid-based**
Centralised *mode*	CSMA MAC [126] IEEE 802.11af [65]	IEEE 802.22 [127]	DSA driven MAC [128] IEEE 802.11af (optional mode) [65]
ad-hoc mode	SRAC MAC [129] HC−MAC [130] DOSS [131] DCA−MAC [132]	C−MAC [133]	OS−MAC [134] POMDP [135] SYN−MAC [136] Opportunistic MAC [137]

The selection of the sensing method should be based on two requirements: first to prevent interference with the PU, and the second one to avoid interference with other SUs. A high-accuracy method is needed more to distinguish PU signals from other competing SU signals. In the time-slotted approach, the SUs share the spectrum based on time-division multiple access (TDMA). The first IEEE standard for CR networks, i.e., IEEE 802.22, is based on this approach. As the mechanism allocates a different time slot to each SU to avoid interference between them [138]. In IEEE 802.11, the sensing operation is mainly needed to protect the PU, as the protocol is not based on contending SUs, and so TDMA networks require less frequent sensing than random access networks.

The hybrid protocols included in the IEEE 802.11af as an optional mode, are used to achieve a compromise between random access and time-slotted approaches and should be reflected in the trade-offs between complex and simplistic sensing methods. However, the expectation that divers CR networks—that is, that use different MAC protocols—will coexist, is another challenge that has an impact on sensing requirements. In such an environment, sensing is required to have the advanced capabilities needed to distinguish between other SU signals and the PU signal, so sensing methods with these capabilities should be used where there are QoS concerns. The use of such advanced methods and its impact on QoS are not considered in most of the CR MAC protocols listed in Table 2.5, including IEEE 802.11af. In addition, several proposed protocols require more than one radio channel for their operation, including DOSS, C-MAC and SYN−MAC, and thus require more complex hardware and spectrum resources. The IEEE 802.22, standard offers the opportunity to use any available sensing technique as long as the protocol regulations, including a maximum sensing duration of 200 ms, maximum P_f of 0.255 and minimum P_d of 0.9, are satisfied [127], but QoS requirements are not considered in the proposed sensing strategies. Nevertheless, the IEEE 802.22 standard is more advanced in terms of utilising sensing techniques than is IEEE 802.11af, which relies on GDB more to protect PU than to sensing, up to the last amendment [139]. The sensing strategies used by CR MAC protocols considered in the literature can mainly be classified as those using a fixed sensing method and those having the ability to select a sensing method during operation, as shown in Table 2.6.

Table 2.6 Sensing strategy classification.

General strategy	Specific Strategy	Examples	Pros	Cons
Fixed sensing strategy	One sensing technique	C–MAC [133] CR-CSMA [140] CR-ALOHA [141] QC MAC [142]	Simple and requiring few resources Easy to analyse its performance	Accuracy depends on the technique used Cannot adapt to changes in operation requirements
	Two or more sensing techniques in series	IEEE 802.22 [127] C2RMAC [143] OMF-MAC [144]	Potential higher accuracy	Longer sensing duration More overhead on the CR device resources
	Two or more sensing techniques in parallel	Fuzzy logic based spectrum sensing [145]	Potential higher accuracy.	More overheads on CR device resources. No significant improvement in accuracy.
Select sensing strategy	Selecting sensing technique	Selection of techniques for obtaining information about spectrum availability [146, 147]	Flexible strategy can be adjusted dynamically based on requirements. Potential higher accuracy.	Selecting mechanism overheads. More overheads on CR device resources.

The fixed sensing strategy uses either a single technique or more than one, without a selection mechanism, to perform spectrum sensing. The simplest fixed sensing strategy is based on a single technique, usually, ED, for a fixed data transmission frame and sensing duration, as discussed in the previous section 2.4. Such an approach is not suitable for random-access MAC protocols that require variable lengths to transmit data frames; nor can an efficient trade-off between PU protection and SU QoS be met by this restricted strategy.

For CSMA, a CR-CSMA protocol based on the use of single sensing technique like ED, with possible different sensing times and data frame durations, is proposed in [140] and later improved in [141]. Because it provides a preliminary analysis of throughput and delay for CR-CSMA, several researchers have used its performance as a benchmark against which to measure their proposed solutions, as in [144, 148, 149].

CR-CSMA was designed to use only ED, which is not suitable for CR because of limitations in its sensing outcomes. As the performance analysis provided for CR-CSMA is of a fixed data frame size and for a maximum sensing duration of 20 ms, more analysis is required for different frame sizes belonging to other applications, and with various sensing durations. Furthermore, any sensing time should be long enough to enable more highly accurate sensing techniques to be conducted. Another protocol called QoS-aware cognitive MAC (QC MAC) for CSMA based networks has been proposed, based on a variable sensing duration strategy, to consider the need of changing the sensing duration according to the QoS requirements [142]. Service differentiation is considered and provided by the sensing strategy used in QC MAC, although it does have some limitations: one is that it is based only on ED. Other concerns regard the proposed service differentiation mechanism, which prioritises only three types of application traffic: voice, video and data. In addition, the QC MAC protocol relies on using a control channel, rather than the data channel, to exchange control information, so more than one communication channel and transceiver are required in its operation.

For advanced sensing, more than one sensing techniques are used in series or in parallel to improve outcomes. The use of two techniques in series is adopted in the IEEE 802.22 [127]. First, Coarse sensing is conducted via a blind sensing technique to detect if the channel is occupied. If no signal is detected, fine sensing is conducted with a more accurate technique based on a signal specific detection method; such methods are described in 2.2.2; the minimum required time for such signal-specific sensing could be up to 98.3462 ms [127]. A longer sensing time is required for better sensing performance in low SNR. For example, correlation sensing requires 387.2 ms under SNR of –14 dB, but the maximum permissible sensing duration is about 158 ms in IEEE 802.22 [127]. Hence, implementing high-accuracy sensing methods that require longer sensing times is not compatible with the standard.

Moreover, the criteria suggested to trigger the use of fine sensing is based only on the outcome of the coarse sensing. As outcome accuracy is subject to P_d and P_f, and poor performance is expected in low SNRs, the probability of conducting fine sensing when it is not actually needed, or vice versa, may increase in channel conditions. However, the suggested mechanism of conducting two stages of sensing, coarse and then fine sensing, is not mandatory, and other criteria and mechanisms that comply with IEEE 802.22 could be implemented. For instance, the use of only fine sensing is found to cause less delay for voice applications than the suggested strategy of using Coarse and then fine sensing [150]. The implication is that the sensing strategy should be based on the QoS requirements of the running application, which has not yet been addressed in IEEE 802.22.

An opportunistic matched filter-based MAC (OMF-MAC) has been proposed for CR networks based on CSMA/CA with DCF [144]. The authors realised the necessity of distinguishing between PU and SU signals, and addressed this issue by including matched filter detection in their sensing strategy. They use ED first, and only if the channel is found to be busy is the matched filter applied to determine if the PU transmission exists in the operational channel. The criteria for conducting matched filter detection are not the criteria suggested in IEEE 802.22, as aforementioned. Also, the expected inaccurate outcome of ED, particularly in low SNR, may still lead to a wrong decision about using the matched filter. The required long sensing time for matched filter detection was not considered in the performance analysis for OMF-MAC, and nor was the impact of using OMF-MAC on the QoS of various possible applications with different requirements.

For using sensing techniques in parallel, three techniques, matched filter sensing, cyclostationary detection and ED, were used simultaneously and the results evaluated, as proposed in [145]. However, conducting more than one sensing method concurrently is computationally expensive, especially when complex techniques are involved such as matched filter sensing, as the required sensing duration and power will have a significant impact on the running applications and CR device resources. Consequently, conducting more than one sensing technique in parallel is rarely adopted because its improvement in accuracy is insignificant compared to using only the most accurate method. Any detection approach using more than a single sensing technique, either in serial or parallel, can be considered as one fixed sensing method with one resulting sensing outcome, as demonstrated in Figure 2.6. The method's accuracy and required duration are based on the techniques and algorithms used. Like the other fixed sensing strategies discussed above, it does not have adequate flexibility to adjust the sensing function according to requirements.

The select sensing strategy provides flexibility by selecting the sensing method that best fits a given situation, using the mechanism shown in Figure 2.7. The term 'sensing method' is used to imply that the selected method could involve more than one sensing technique. Hence, any of the described fixed sensing strategies may be employed in the select sensing strategy when appropriate. In such a strategy, a wide range of implementation options could be designed based on the selection mechanism, its input parameters, and the range of output sensing methods. This approach has not received sufficient consideration yet. Based on the reviewed literature, only in three articles by a small set of allied authors has the select sensing strategy approach has been adopted [146, 147, 151]. The authors proposed a selection mechanism based on four parameters: the required probability of detection, operational SNR, available time for performing the detection, and available prior information [146]. The QoS requirements of different possible applications in a CR system are not directly considered in their selection criteria. Moreover, the selection mechanism is not applied to a specific MAC protocol, and its performance is not evaluated.

Six possible outputs were used in the proposed selection mechanism in [146]. Five of the six outputs are sensing methods, including, ED, cooperation sensing based on local ED, correlation, feature and matched filter detections. The sixth output is an action of changing the operation channel. Changing the channel decision is related to spectrum management and mobility functions in CR as discussed in Section 1.1. These input parameters are not adequate for deciding on the use of cooperative sensing or changing the channel. The cooperation between SUs depends on the available channel, algorithms and MAC structure for exchanging local sensing outcomes. Both ED and cooperative ED outputs will result in the same action regarding the nominated technique for local sensing. Of these sensing strategies, the select strategy is the most promising approach for enhancing the CR performance. Nonetheless, more care and effort is needed to design a sensing selection mechanism, with related inputs and outputs.

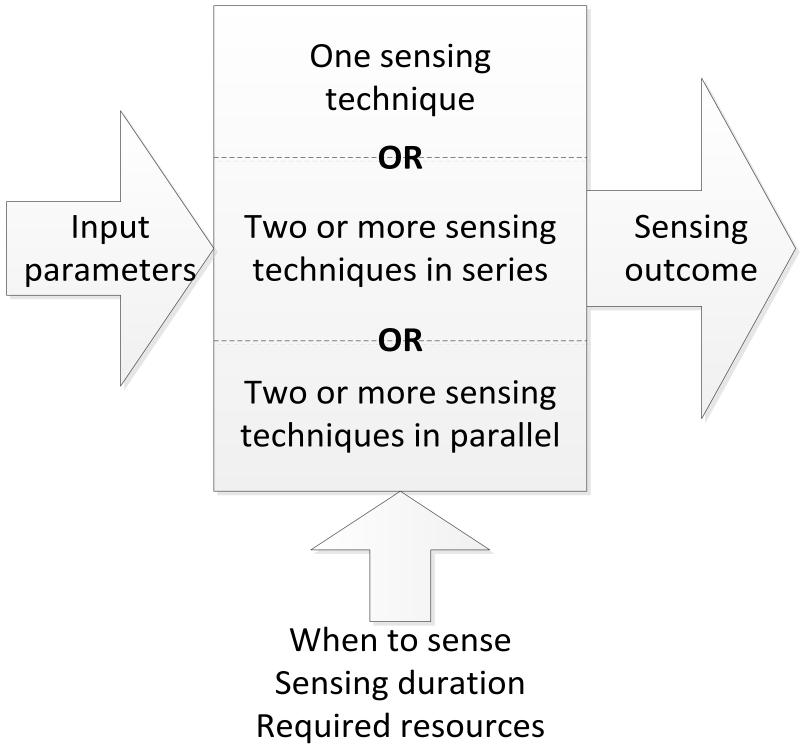


Figure 2.6 Fixed sensing strategies

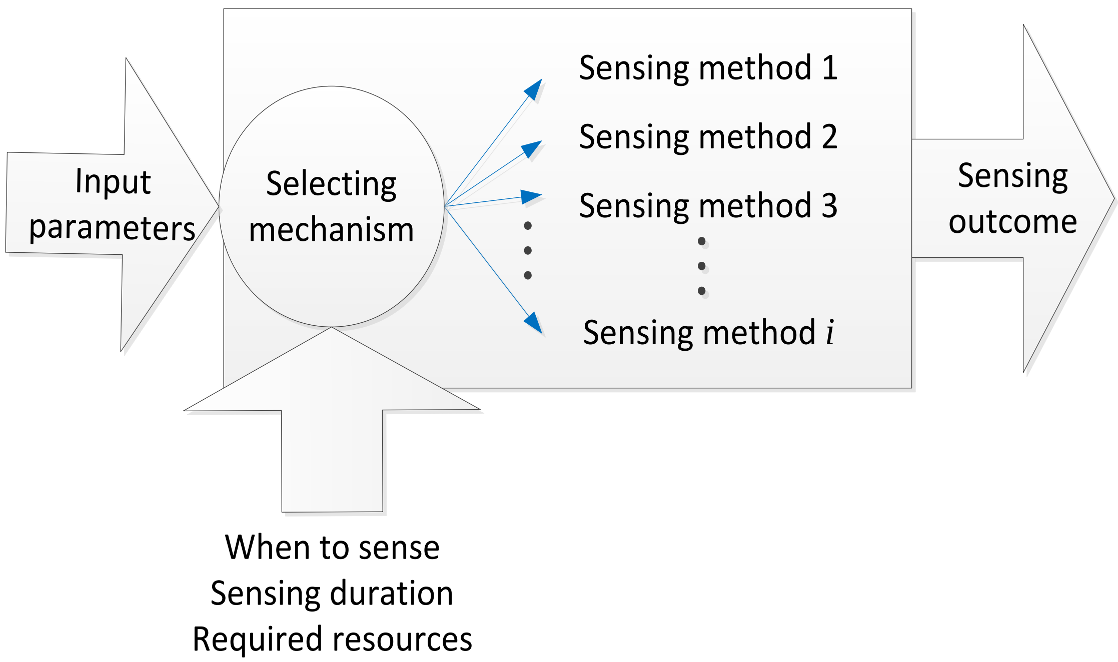


Figure 2.7 Selecting sensing strategy

2.6. Summary and discussion

The variety of sensing techniques discussed in this chapter demonstrates that distinct advantages and drawbacks are present in all. For instance, the simplest sensing method, ED, needs less power and time compared to other sensing methods, and thus is widely adopted in studying CR performance. However, it has low sensing accuracy, particularly in low SNR. Moreover, it can only detect the busyness of a channel, and cannot determine if it is because of PU transmissions or not. Such limitations result in poor spectrum utilisation and PU protection. Advanced sensing techniques are needed, but they require more resources.

The key findings from investigating and comparing sensing techniques characteristics are:

1. Blind sensing methods, such as ED, cannot distinguish PU from SU signals.
2. Sensing methods based on prior information about PU signals, such as matched filter, can distinguish PU signals from other signals, but require much longer sensing durations than blind sensing methods.
3. A longer sensing duration results in more accurate results per channel.
4. For each sensing method, its accuracy (P_d and P_f) will be affected by the SNR of the sensed channel. The SNR may change unpredictably during operation.
5. The sensing outcome level and its accuracy differ from one sensing method to another under equivalent channel conditions (such as having the same SNR).
6. Each sensing method has a critical SNR, the SNR wall, below which the sensing accuracy cannot meet the sensing requirements; i.e., $P_d > 0.9$ and $P_f < 0.1$.

The level of cognition about the surrounding RFS that a sensing technique achieves is an essential factor in the CR paradigm. In this research, the outcomes from a sensing operation are classified into three levels (see Table 2.1). The third level represents the most advanced knowledge about the RFS that a sensing operation can collect from its surroundings. In the future, more levels may be added depending on the extra details that may be accumulated by sensing techniques. Advanced sensing outcomes with high accuracy, allow a CR system to exploit spectrum holes more efficiently, but at the cost of overheads, particularly, in terms of time. For instance, a sensing time longer than 300 ms is required for correlation sensing to achieve higher accuracy of detecting PU signals among other signals, and the required sensing time must be multiplied by the number of channels to be assessed. Moreover, the sensing has to be conducted more frequently than any other CR function during operation, so the operation has a critical impact on the QoS of the running applications and overall performance in CR. This impact varies depending on the sensing technique and strategy used.

By reviewing the available sensing strategies, it has been found that more research is required on using high-accuracy sensing that meets the various QoS requirements. Using one sensing method is not suitable for all CR operations. Likewise, using more than one sensing technique, either sequentially or simultaneously, offers no significant improvement over the use of a single high-accuracy method. Research attention should therefore be drawn to dynamic sensing strategies that change according to operation requirement changes. In particular, a strategy should select the proper sensing technique, in real time, based on any change of QoS requirements. This study shifts the focus to a new approach, where a CR device supports a range of various sensing methods, allowing the proper sensing technique to be selected based on real-time requirements. This approach implies the need for a real-time mechanism to select the most suitable sensing method based on a number of factors or criteria. Such factors are considered, and a real-time selection mechanism is proposed, in the next chapter.

The major research gaps identified in this chapter are addressed in this thesis as follows:

- Studying the impact of long sensing duration, up to 500 ms, see Section 5.3).
- Various accuracies about QoS (see Section 5.4).
- Proposing a select sensing strategy that considers QoS requirements among other essential factors, with a minimum overhead of the selection mechanism (see Section 3.5).
- Applying the proposed select sensing strategy to CR networks based on CSMA/CA and designing a novel mechanism integrated with IEEE 802.11 standards for enhancing QoS (see Section 4.4).

Chapter 3. Sensing selection solution for QoS improvement

The different sensing methods analysed in the previous chapter each have advantages and shortcomings for particular scenarios dealing with cognitive radio (CR) operation. In particular, sensing methods can greatly affect QoS for users of cognitive radio networks, leading to the conclusion that no particular sensing technique can be adjusted to operate efficiently for all CR devices' operational situations and requirements. In this chapter, this issue will be addressed by proposing a fuzzy logic decision-making scheme for selecting the proper sensing method from the catalogue of those available. The fuzzy logic modelling toolbox in MATLAB 2014a will be used to implement the proposed decision-making scheme.

This chapter starts with a brief introduction in Section 3.1. Related work and motivations for proposing this solution are presented in Section 3.2. Key factors for selecting the suitable sensing methods are identified and discussed in Section 3.3. Background to the fuzzy logic approach and its implementation strategies are provided in Section 3.4. In Section 3.5, the proposed fuzzy logic-based QoS-aware scheme for selecting the proper sensing method is explained in detail. A summary and discussion of this chapter are presented in Section 3.6.

3.1. Introduction

CR technology relies on the information that can be gathered about the surrounding radio frequency spectrum (RFS). Detecting the primary user's (PU)'s presence, mainly to find out if the PU is using the spectrum band or not, is the major activity of a CR. The accuracy of the operation is measured by the probability of having achieved accurate detection and a reduced probability of a false detection of the PU's presence [24]. The nature of electromagnetic signals makes accurate sensing a complicated operation. In particular, the uncertainty of noise level and signal-to-noise ratio (SNR), and the multipath fading of PU signals can significantly affect sensing accuracy [28, 29]. The hidden PU problem is another issue that occurs when an SU cannot detect the presence of a PU who is actually within the same transmission range [30]. The challenge that arises is to implement a flexible sensing function to cope with the rapidly changing characteristics of the radio environment and the varied PU systems [74].

Selecting the appropriate sensing function is also critical, as the sensing can affect the error rates for both the primary and secondary systems [37]. As such, it is imperative to investigate the impact of the sensing operations on QoS levels. The results of such investigations, reported in Chapter 6, are utilised to identify ways to mitigate such impacts. The outcome is proposed as a fuzzy decision-making scheme for selecting the proper sensing method. First, the proposed solution should be simple to minimise its overhead. Second, the system inputs should be limited to the most important parameters to compromise between efficiency and simplicity. Third, the input and output of the decision-making scheme should be generalised so that the solution can be used in a broad range of applications and sensing methods.

3.2. Related works and motivations

From the early stages of developing CR technologies, several researchers have considered the correlation between the sensing operation and QoS levels, or some other relevant form of metric, in their studies. For instance, the relationship between the sensing duration and the CR throughput has been reported in [110, 111]. These works focus on optimising sensing duration to achieve maximum throughput. However, they only use the energy detection (ED) method for local sensing, and the results cannot be generalised to sensing methods, that require different sensing times and resources. ED also has poor sensing accuracy compared with other sensing methods. Another factor of a sensing operation that may affect the QoS level is frequency; i.e., how often the operation will be conducted. The first reported research on the impact of this factor on the QoS of multimedia applications, using a mathematical model to determine an optimal sensing frequency in CR networks, is found in [108]. Such studies can show improvements for one or more of QoS requirements for a certain sensing method within a limited range of spectrum characteristics such as SNR levels, fading, and PU behaviour.

Reviewing several sensing techniques and strategies in relation to QoS has been reported in Section 2.2 and Section 2.5. In this section 3.2, the focus is on related works that have developed fuzzy logic algorithms in the sensing function. To improve sensing accuracy, three techniques known as matched filter sensing, cyclostationary detection, and ED, are used simultaneously, and their results are evaluated by a fuzzy logic algorithm as proposed in

[145]. However, this is computationally expensive. In particular, when complex techniques are involved, such as matched filter sensing, the required sensing duration and power will have a significant impact on running applications and CR device resources. Another approach also uses fuzzy logic to improve sensing accuracy by evaluating the results of two sequential sensing stages, the first with ED and the second with cyclostationary detection, as proposed in [152]. Although the two-stage approach requires less overhead than the three concurrent sensing model, the two sensing techniques are unable to distinguish PU from SU transmissions.

The sensing methods discussed so far have different advantages and drawbacks, and none is suitable in all CR operation scenarios. For instance, the simplest sensing method, ED, needs less power and time than other sensing methods, but has low sensing accuracy, particularly in low SNRs, that results in poor spectrum utilisation. A more advanced sensing method is required, one that uses more power and time for sensing. This thesis is based on the argument that a CR device has to be capable of using various sensing methods so it has a broad range of sensing capabilities. It should be capable of using the proper sensing method dynamically, based on such operating requirements and constraints as those imposed by QoS requirements and CR device capabilities. The only known study applying a similar approach is [146]. Its focus is on four parameters: the required probability of detection, operational SNR, available time for performing the detection, and available a priori information. The QoS requirements for different applications in a CR device, considered a vital requirement in this current study, are not directly considered. Obtaining full knowledge of the current SNR level is impractical since the noise power changes unpredictably with location and time, making it unfeasible to rely on this parameter to select sensing methods in real time.

In this thesis, QoS requirements are involved in selecting the proper sensing method. The key factors that should be considered for a more comprehensive and practical solution are also investigated, then a set of fuzzy logic rules that can be generalised to different applications and sensing methods is designed. A CR design requires a performance benchmark with established metrics, but these are not yet available [153]. Instead, a basic benchmark for performance metrics directly affected by sensing in a CR paradigm is suggested in this thesis, based on recognised performance metrics for existing wireless networks.

3.3. Factors for selecting a suitable sensing method

Selecting a suitable sensing method for a particular CR operation and its particular conditions depend on several factors that may change during the operation. This suggests that an optimum sensing method for the new conditions might need to be designed. In this study, five key factors with a noticeable influence on selecting the sensing method are identified. Most of the aspects thought to affect the selection of the most appropriate sensing method fall under one of these: the application QoS requirements, the available information about the PU signal and RFS, the required protection for the PU, the CR device capabilities, and the CR network mode and capabilities. These are discussed in the following subsections.

3.3.1. Application QoS requirements

QoS requirements differ depending on the applications running on a CR device, and different sensing methods and their parameters have different impacts on these requirements. As discussed in Section 2.4, the required sensing time and the accuracy level vary depending on the sensing method used. For example, the sensing time for ED is usually less than 1 ms while cyclostationary feature detection requires 24 ms or more [72]. Using a sensing method with high accuracy leads to greater utilisation of the spectrum. However, it requires more sensing time and causes more delay to the running application. The sensing delay and transmission throughput vary from one sensing method to another within the same operating conditions. As a result, the sensing operation used on a CR device has a direct impact on the QoS of an application running on the device, mainly in terms of the throughput, delay and jitter. As sensing is a repetitive operation, a CR device should be able to select a suitable sensing method with the least impact on the QoS of the application. For optimal sensing accuracy, the sensing time should be in the range of 2 seconds [154]. However, such a long time is not suitable for delay sensitive applications such as voice over IP (VoIP), in which case sensing methods that require long times should be avoided. Other operational requirements must also be taken into account.

Increasing the sensing duration and conducting sensing more frequently will improve the probability of correct detection of the PU's presence. While this leads to more protection to the PU from interference by CR users, it leads to QoS degradation for CR users. The degradation can be measured by several parameters, such as throughput, delay, and medium access control (MAC) layer process overheads [108]. Determining the sensing time and frequency of sensing needs to make a trade-off between protecting the PU's QoS and improving the QoS of CR users, and the level of protection required may vary with the frequency bands and types of services involved. QoS requirements should be included among the other factors taken into account when selecting a sensing method.

3.3.2. Available information about the PU signal (prior information)

The amount of information available about the characteristics of the PUs and the communications media is an important factor influencing the selection of a proper sensing method. For instance, insufficient information about PU signals excludes the use of methods that require such prior information, such as matched filter sensing. To achieve optimal sensing accuracy, a CR device should be able to change the sensing method based on information that becomes available about the PU signal. ED performs more effectively when the SNR is high, and ED could be the preferred sensing method in this situation because it requires minimal sensing time, power consumption and hardware complexity. However, ED's measurement of the power level can only indicate if the sensed channel is busy or not. It is hard to tell if the channel is busy because of PU transmissions or other transmissions. Where there is low or uncertain SNR, ED performs very poorly and should be avoided. In the absence of any prior information about the PU, the choice of a sensing method is limited to blind sensing methods. The range of sensing methods based on prior information that can be used is dependent on the type of available information about the PU. For example, the information about the transmission technology used by a PU can be enough for sensing methods based on radio identification, but not for those based on the use of matched filter sensing, which require more detailed information. Obtaining very specific information about a PU, including the modulation used and frequency of operation, is not possible in some cases. The operation of a CR device usually involves switching between different frequency bands licensed to different PUs, so that the CR device has to be able to change its sensing method based on the information available about the new frequency band.

3.3.3. The required protection for the PU

In general, PU transmissions in licensed bands must be protected from interference by CR signals. However, different PU signals, depending on their technologies and frequency bands, may have different robustness against interference. PU signals with higher robustness to interference require a lesser level of protection. Simply, the required protection can be considered high or low. The PU signals that require high protection will force a CR user to use a high-performance sensing method, while those that require low protection allow the CR more flexibility in selecting a low-performing sensing method.

The degree of protection required may vary depending on the frequency band used and the type of service provided. For instance, analog TV is more robust against interference than digital TV [109]. Therefore, a sensing method that provides less protection, like a lower PU detection probability, may be used when PU transmissions belong to analog TV services. The PU application's sensitivity should also be considered when determining the protection level required. For example, PU signals used for military and security purposes require a high level of protection. High-accuracy sensing may be enforced in frequency bands where PU signals are used for critical missions. This poses the need to classify different protection levels for PUs based on the transmission technology they use and their sensitivity. These protection levels can be assigned to each licensed band, and announced.

This factor is related to a requirement outside the SU system, so must be provided by the PU system or an authorised body such as a government organisation responsible for regulating frequency bands. Currently, IEEE standardisation for CR specifies a general protection level regarding the sensing technique being used by SUs; the minimum acceptable level of detection probability is 90% [155]. Other protection criteria such as setting an acceptable level of a PU's loss rate, or interference power caused by SUs, are also proposed [156]. In this chapter, the focus is on the protection criteria directly related to selecting which sensing method to use. The wide adoption of CR technology is based on how successful it is in guaranteeing an adequate level of protection to PUs.

3.3.4. CR device capabilities

A CR device designed with limited hardware resources and power capacities will not be able to support a wide range of sensing methods. Some methods require sophisticated hardware components and high power consumptions, as in the matched filter sensing method, compared with simpler methods such as ED. An ideal CR device should be able to be reconfigured on the fly to support a broad range of sensing methods. In practice, a CR device's actual capability will limit the array of sensing methods that can be supported. For a mobile CR device capable of supporting several sensing methods, the available battery power can be considered by selecting a sensing method requiring lower power consumption.

3.3.5. The CR network mode and capability

The network mode and capability are important factors to CR systems when deciding between cooperative and non-cooperative sensing approaches. In CR networks with infrastructure and centralised topology, a method based on cooperative sensing is more suitable than one based on local sensing only. Hence, the capability of such a CR network depends on how much control ability can it provides for determining white spaces. Its capacity also relies on how much information it can gather and provide to its CR users about PU signals and the ambient spectrum.

3.4. Fuzzy logic approach for decision-making

As there are different sensing methods, each with advantages and compromises, and it is difficult to select one as the optimal solution for all CR applications and scenarios, it is necessary to have a selection mechanism to choose the most effective sensing method based on the factors described in 3.3. As these factors may change in real time, the selection mechanism should be able to handle real-time changes and react appropriately. The selection mechanism can be considered a real-time decision-making system to choose among possible alternative actions to achieve one or more goals. The output decision of such a system depends on available accurate input, a base knowledge of the matter at hand, and the extent to which the system is intelligent. Fuzzy logic provides an appropriate mathematical tool to make decisions in situations where inputs are vague and imprecise, or may be qualitatively interpreted. Particularly, in this study, the available information about the decision-making inputs tends to be heterogeneous and qualitative. Fuzzy logic can transform the input information into homogeneous membership values, which can then be processed through a set of fuzzy inference rules [157]. The resulting model, technically a fuzzy interference system (FIS), is simple and fixable, and can easily be implemented with minimal impact on the system's resources. In the following subsections, the fuzzy logic concept and the FIS are explained in more detail.

3.4.1. Fuzzy logic

The concept of fuzzy logic was introduced by Lotfi A. Zadeh in 1965 [158]. Zadeh invented fuzzy set theory, describing mathematically the fuzzy logic concept based on classical set theory. Conventionally, set theory is used for crisp or classical data, such as an element, which can be considered either a member or not a member of a certain defined set (or class). For example, the number '1' is a member of integer numbers while 'half' is not. In contrast, fuzzy set introduces the concept that membership of an element may be represented as a partial degree of membership. For example, for the set of real numbers that are greater than one, its element membership can be defined as a grade that increases as the number is larger than one. Similarly, when dealing with vague terms, 'somewhat satisfied' may have a different membership degree in both 'satisfied' and 'unsatisfied', predefined classes. Usually the degree of fuzzy set membership is denoted as between 0 and 1 for an element. The value zero means that the element is not a member of the fuzzy set, while the value one means that the input element is a full member of the fuzzy set. The notions, such as of inclusion, union, intersection, complement, relation, convexity, are mathematically represented in classical set theory, and fuzzy set theory allows them to be extended and used in fuzzy sets.

In 1974, Ebrahim H. Mamdani proved the worthiness of applying fuzzy logic to an automatic control strategy for a a steam engine [159]. His name was given to one of the well-known ways of implementing FIS, described in Section 3.4.3.1. Computer systems can be implemented to deal with crisp data, but fuzzy data cannot be handled by machines unless it is represented as crisp data. For example, fuzzy terms that humans can understand, such as 'high', 'medium' and 'low', are not understandable by machines unless they are represented in a mathematical form. The fuzzy logic concept allows conversion between fuzzy and crisp data, so that machines can deal with control/decision/selection matters in ways similar to human logic. In other words, 'making computers think like people', as Mamdani's titled one of his publications [160]. Gradually, fuzzy logic has been widely adopted and applied successfully in many different real-world applications [161]. Fuzzy logic techniques convert human language rules and data to their mathematical equivalents so they can be easily implemented in computer systems. This simplifies designers' efforts to provide flexible solution models for complex systems dealing with imprecise and incomplete data.

The fuzzy logic model FIS is composed of a number of conditional 'If-then' rules that deal with human language terms. A designer who has a knowledge base of the system can set as many rules as required, although realistically a moderate number should be used for modelling real-world systems. Typically, implementing a FIS for an application goes through three main phases [162] (see Figure 3.1).

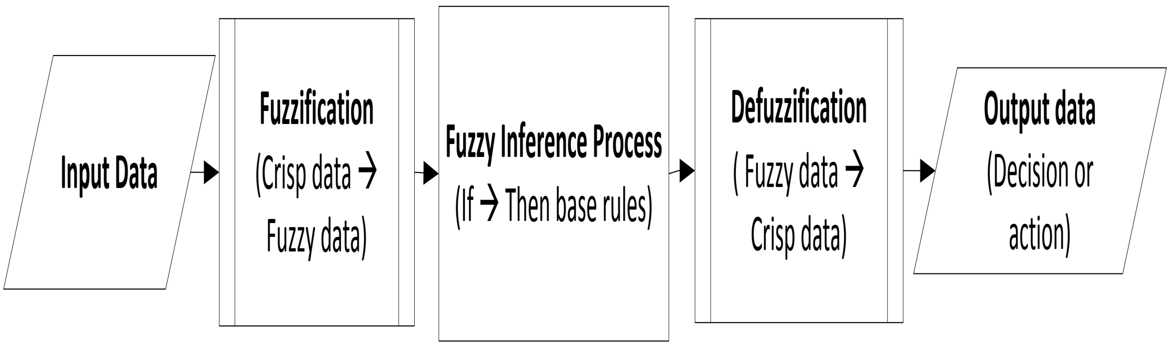


Figure 3.1 Fuzzy inference system

Phase 1: **Fuzzification** converts classical or crisp data into fuzzy data based on its membership function (MF). In fuzzy set theory, an MF defines the relationship between the value of an element and its degree of membership in a fuzzy set [160]. It is a graphical representation, or mathematical function, defining how each point in the input space is mapped to a degree of membership, between 0 and 1, in a certain fuzzy set.

Phase 2: The **Fuzzy Inference Process** combines membership functions with the 'If-then' base rules to derive the fuzzy output. This step uses fuzzy operations based on fuzzy set theory to derive a fuzzy output from the fuzzy input in a given FIS.

Phase 3: **Defuzzification** uses different methods to calculate a crisp value from the fuzzy output of the fuzzy inference process.

These phases are discussed in the following subsections.

3.4.2. Fuzzification

In this phase, each input variable is translated to a graphic form demonstrating its degree of belonging to the appropriate fuzzy set. The graphic representation of each input set, or class, is the MF. The crisp input data belong to a specific class represented in the *x*-axis, and its membership degree is shown in the *y-axis*. Any arbitrary curved shape can be used for the MF as long as it remains between 0 and 1. However, the MF should be simple and efficient, so its graphical analysis imposes minimal overheads on the system. For a real-world scenario, the designer decides how many MFs are required relative to the input variables defined for the system, and chooses a name for each input variable set and each element in the input sets, all in human-friendly language. For example, the input set name could be 'Application QoS', referring to the required QoS variable of the different applications running in a system. The element names of the 'ApplicationQoS' set could be 'Low', 'Medium' and 'High' for three predefined QoS levels at which applications may run on the system. The given names are understandable by humans, and logic analysis can be applied to them, but computer systems need a mathematical representation of them. The designer can define the MF set by using different graphs.

The MATLAB fuzzy logic designer toolbox is used to demonstrate four popular MF graphs, also called MF types, as shown in Figure 3.2, Figure 3.3,Figure 3.4 and Figure 3.5. Other graphs can be used, as mentioned before. The toolbox provides eleven built-in MF types that can be further customised if needed; the designer can use these built-in shapes or create a customised MF as required.

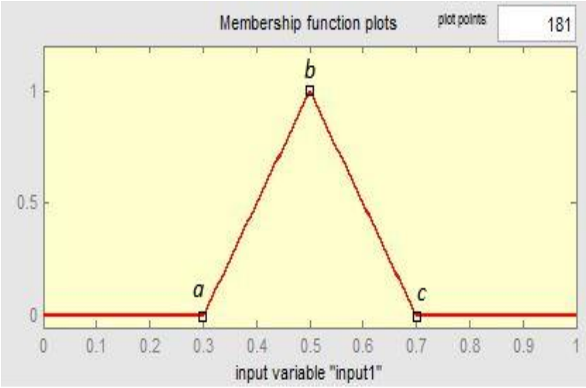


Figure 3.2 Triangular MF (trimf)

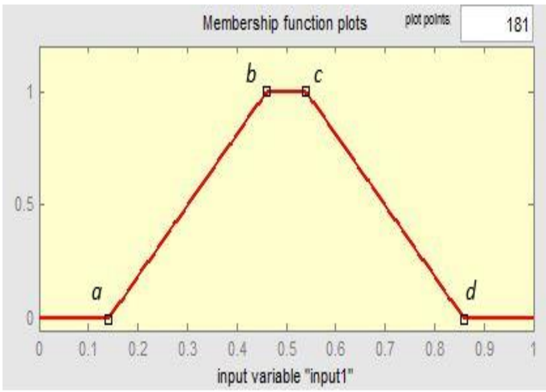


Figure 3.3 Trapezoidal MF (trapmf)

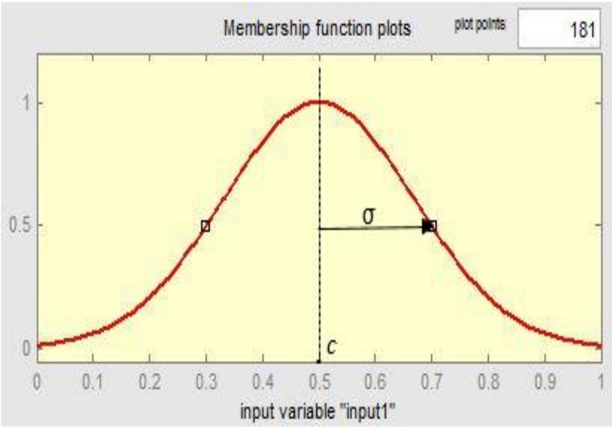


Figure 3.4 Gaussian curve MF (gaussmf)

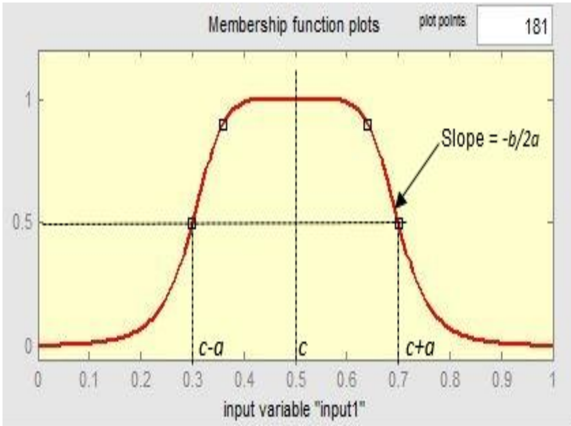


Figure 3.5 Generalised bell-shaped MF (gbellmf)

The simplest membership function is the *triangular* membership function, named *trimf* in MATLAB, which is shaped by using straight lines and three points, *a*, *b* and *c*, forming a triangle, as shown in Figure 3.2. The parameters *a* and *c* are called the feet; i.e., the base of the triangle shape, and *b* is the peak as given by:

$$f(x;a,b,c) = \begin{Bmatrix} 0, & x \le a \\ \dfrac{x-a}{b-a}, & a \le x \le b \\ \dfrac{c-x}{c-b}, & b \le x \le c \\ 0, & c \le x \end{Bmatrix} \qquad (3.1)$$

The *trapezoidal* membership function, *trapmf* in MATLAB, is a truncated triangle curve with four points, *a*, *b*, *c* and *d*, connected by straight lines as shown in Figure 3.3. The equation is defined by these four parameters:

$$f(x;a,b,c,d)=\begin{cases}0, & x\le a\\ \frac{x-a}{b-a}, & a\le x\le b\\ 1, & b\le x\le c\\ \frac{d-x}{d-c}, & c\le x\le d\\ 0, & d\le x\end{cases} \tag{3.2}$$

The MF graphs used in Figure 3.4 are called *gaussmf* as they are built on the Gaussian distribution curve. The symmetric Gaussian function is defined by two parameters; σ and c, as given by:

$$f(x;\sigma,c)=e^{\frac{-(x-c)^2}{2\sigma^2}} \tag{3.3}$$

Similarly, generalised bell MFs, shown in Figure 3.5 and named *gbellmf*, are based on a Gaussian distribution curve (non-linear) but with a flat top, so it has a bell shape and extra parameters compared to *gaussmf*. The generalised bell function depends on three parameters *a, b,* and *c*:

$$f(x;a,b,c)=\frac{1}{1+\left|\frac{x-c}{a}\right|^{2b}} \tag{3.4}$$

The straight-line (linear) MFs, such as *trimf* and *trapmf*, are preferred for their simplicity and low-cost execution in terms of resources and time: both necessary features in systems with limited resources and time-sensitive applications. In contrast, non-linear MFs, such as *gaussmf* and *gbellmf*, are appropriate because of their smooth and nonzero input-output relationship. However, they involve calculations using exponential functions that require more computations and power. Such MF types that have symmetric shapes are not suitable for applications with asymmetric input data, but their symmetric graphs can be modified to adapt to this requirement, or asymmetric MF shapes can be used. The designer should consider the features of the different MF shapes when defining the fuzzy output sets.

3.4.3. Fuzzy inference process

The core FIS process accrues in this phase. The MFs are combined with fuzzy control rules to derive the output desired of the FIS. The fuzzy control rules are built on If-then rule statements that are predefined by the designer, based on experience or knowledge of the system. A single fuzzy If-then rule has the following format:

if input *x* is *A* ***then*** the output *y* is *B*

where *A* and *B* are linguistic values defined by fuzzy sets.

The ***if*** part of the statement is the antecedent or premise, while the ***then*** part is the consequent or conclusion. In the antecedent, 'is' specifies the current value of the input variable (resulting from fuzzification). In contrast, 'is' in the consequent assigns the *y* to the entire fuzzy output set *B*. In MATLAB terms, different operators tend to distinguish between a relational test in the antecedent by using '**==**' and a variable assignment in the consequent by using the '**=**' symbol. The rule can be expressed as follows:

$$\overbrace{\textbf{If}\text{ input }\mathbf{==}\text{ A,}}^{\text{Antecedent}}\ \overbrace{\textbf{then}\text{ output = B}}^{\text{Consequent}}$$

where A, B are fuzzy set elements defined by human language terms

As the antecedent may consist of more than one input, the fuzzy operations '***and***' or '**or**' can be used to combine the input variables. In addition, the operator '**not**' can be used to negate, logically, an input variable. The If-then rule can be generalised as below:

If input 1 **==** A_1 **and/or** input 2 **==** A_2 **… and/or** input n **==** A_n , **then** output = B_i

where *n* is the number of input variables. Each of the values, A_1, A_2 **...** A_n is an element of the MF corresponding the input variable. B_i is one of elements of the fuzzy output set. The system may have more than one output set; and if so, each rule may have more than one output.

The fuzzy operator ‘and’ can be conducted in MATLAB by selecting one of the following methods:

- Minimum ‘min’: combines the inputs in the antecedent by taking the minimum MF degree of the input values
- Product ‘prod’: combines the inputs in the antecedent by calculating the product of the MF degrees of the input values

For the fuzzy operator ‘or’, one of the following methods can be used:

- Maximum ‘max’: combines the inputs in the antecedent by taking the maximum MF degree of the input values
- Probabilistic OR (also known as the algebraic sum) ‘probor’: combines the inputs in the antecedent by calculating the probabilistic OR of MF degrees of the input values. For example, if the input has two elements with A1 and A2, then A1 or A2 = A1 + A2 - A1 A2

When ‘not’ is combined with an input, e.g., A_1 then **not** A_1 = 1 - A_1. The same operation applies if ‘not’ is combined with the output B; i.e., **not** B = 1 –B. In other words, the input combination of an If-then rule is used to find the fuzzy output element of this rule. For a precise design, the fuzzy If-then rules must cover all possible combinations of fuzzy set input values resulting from the first step, fuzzification. All these rules are assessed in parallel, regardless of order, to produce one fuzzy output set for the current input values. Usually the rules are assigned the same weight, typically 1, but other values between 0 and 1 can be assigned to have rules with different influence in the rule evaluation process. The fuzzy output set results from an aggregation process that may use one of the two main FIS types, Mamdani or Sugeno, described in the following subsections. The aggregated output set will be defuzzified in the last step, defuzzification, described in Section 3.4.4.

3.4.3.1. Mamdani-type fuzzy inference system

The Mamdani type is the first FIS introduced in 1975 by Ebrahim Mamdani [159]. The implementation of the Mamdani type involves defining the expected output as a fuzzy set represented in the most suitable MF graph. The MATLAB fuzzy logic toolbox sets Mamdani as its default FIS, and most of the available results of fuzzy systems are based on this type [163]. The Mamdani inference process involves three sub-processes: combination, implication and aggregation, as shown in Figure 3.6. An example of the Mamdani FIS with two inputs and two rules is illustrated in Figure 3.7.

The combination is made by applying fuzzy operator 'and' or 'or' in the antecedent of each rule as described before. The implication is conducted for each rule to re-form the MF of the consequent part according to the rule strength resulting from the combination of the antecedent parts. In MATLAB, two implication methods can be used as follows:

- Minimum 'min': Trims the MF of the consequent at the resulting value of the antecedent part: in other words, the output of each rule is determined by clipping the output MF of the rule relative to its input combination strength.
- Product 'prod': scales the MF of the consequent by the resulting value of the antecedent part.

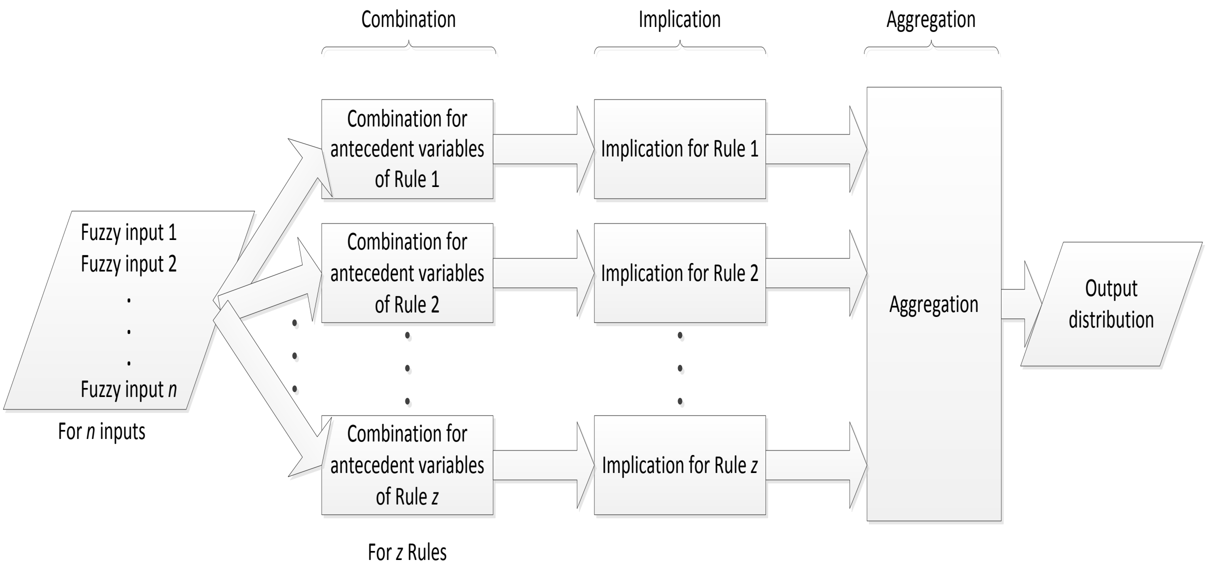


Figure 3.6 Mamdani fuzzy inference process

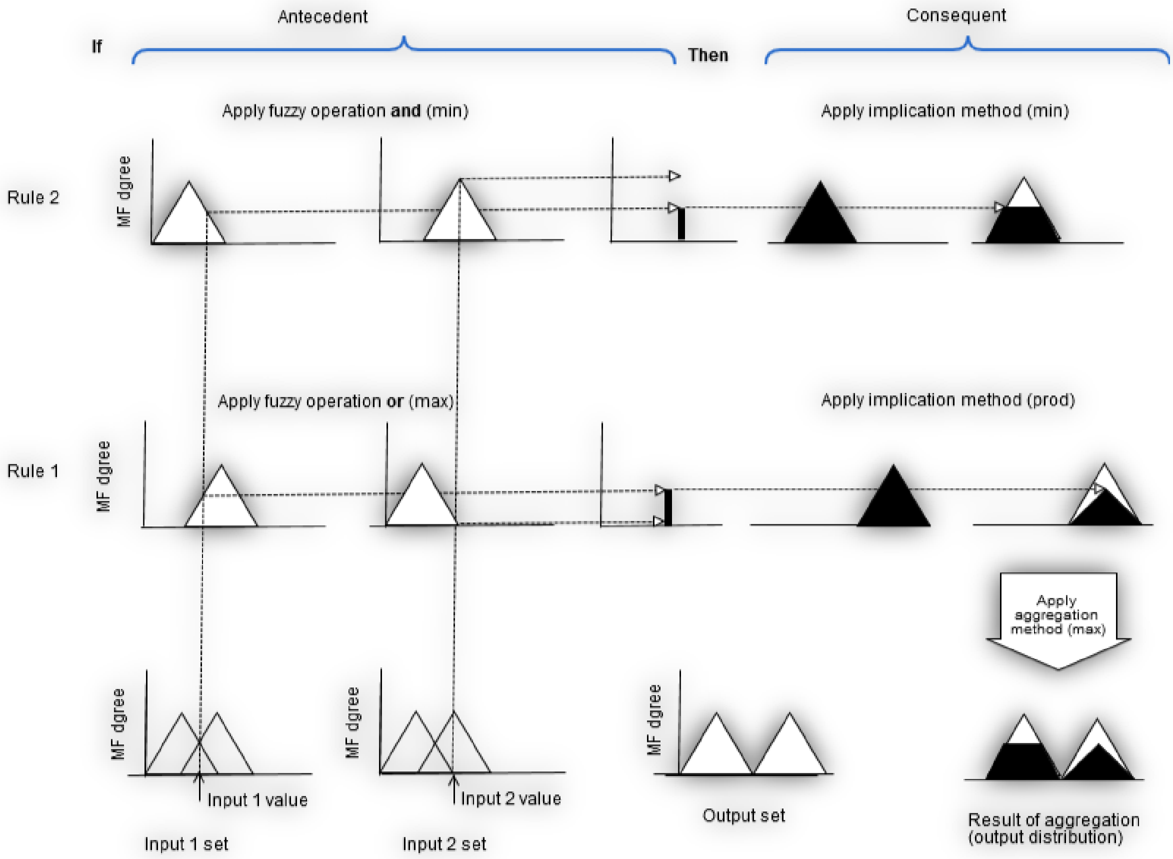


Figure 3.7 An example of a Mamdani type with two inputs and two rules

The MATLAB fuzzy logic toolbox allows users to use other customised implication methods. In the aggregation sub-process, the outputs of all rules are combined to obtain one fuzzy output distribution for the given input values, reshaping the output distribution according to the changes in the input values. The MATLAB fuzzy logic toolbox supports three built-in aggregation methods:

- Maximum 'max': aggregates all rules output sets by taking the maximum MF degree of their fuzzy sets
- Probabilistic OR 'probor': aggregates all rule consequents by calculating the Probabilistic OR of the MF degrees of their fuzzy sets.
- Sum 'sum': aggregates all rule consequents by simply summing the fuzzy output sets of all rules.

The result of aggregation is a fuzzy set distribution, as shown in Figure 3.7. If a crisp output is needed, this is conducted in the defuzzification step, described in Section 3.4.4.

3.4.3.2. Sugeno-type fuzzy inference system

The Sugeno or Takagi-Sugeno-Kang FIS type, introduced in 1985, uses the same Mamdani method for fuzzifying the input, fuzzification, and forming the fuzzy rules, but it uses different methods to determine the consequent [164, 165]. Thus, the procedure of output aggregation and defuzzification is slightly different. Unlike the Mamdani type, the fuzzy consequent in the Sugeno type is calculated as a linear function of the input variables and predefined constants; i.e., the coefficients of the linear function. For instance, in Sugeno type, the If-then rules can be defined as following:

If input **==** X **and/or** Y, **then** output **==** aX + bY + c,

where X and Y are the values of the input fuzzy sets while a, b and c are user-defined constants. In the case of $a = b = 0$, the output will be a constant c. In Sugeno, the output of each rule is a crisp value, calculated either by a linear function or simply a constant. The aggregation output of all rules is then computed by a mathematical combination of the rule strengths and rule outputs. The resulting output is not a fuzzy set distribution, as in Mamdani. In the MATLAB fuzzy logic toolbox, the '**prod**' implication method and '**sum**' aggregation method are mandatory for Sugeno FIS. One of the noticeable problems in designing a Sugeno FIS is the lack of a systematic method for choosing the coefficient values; i.e., a, b and c, used for the consequent part when defining the If-then rules [163]. Moreover, the Sugeno FIS produces only crisp output, while the Mamdani FIS has a fuzzy output that can be defuzzified to crisp value.

3.4.4. Defuzzification

Defuzzification is the last phase in an FIS when one crisp value is required from the aggregated output of the fuzzy rules. In this process, all the fuzzy inputs are defuzzified to calculate one crisp output. For the Mamdani type, the MATLAB fuzzy logic toolbox supports five methods for defuzzifying the distributed output as follows:

- Centroid: this returns the centre of distributed output. The vertical line of the centre value divides the shape of the aggregate set into two equal masses. This method is the recommended and default in MATLAB.

- Bisector: in this method, the vertical line of the returned value will divide the distributed area into two equal areas. In some output distributions the centroid and bisector line are identical; i.e., same output crisp value.
- Smallest of Maximum (SOM): this returns the smallest value for which the distributed output is maximum.
- Middle of Maximum (MOM): This calculates the mean of the values for which the distributed output is maximum.
- Largest of Maximum (LOM): This method returns the largest value for which the distributed output is maximum.

MOM, SOM, and LOM return the same value when the aggregated membership function has a unique maximum; i.e., the peak of the distributed output is not a plateau. An example of applying these five defuzzification methods is illustrated in Figure 3.8, adopted from the MATLAB website [166]. Also, there are many other defuzzification methods proposed in the literature that could suit special requirements [167]. For the Sugeno type, defuzzification is used to calculate a crisp value from the aggregated output of all FIS rules. The MATLAB fuzzy logic toolbox provides two methods, the weighted average '**wtaver**' and the weighted sum '**wtsum**'. **Wtaver** calculates the weighted average of all rule outputs while **wtsum** returns the weighted sum of all rule outputs.

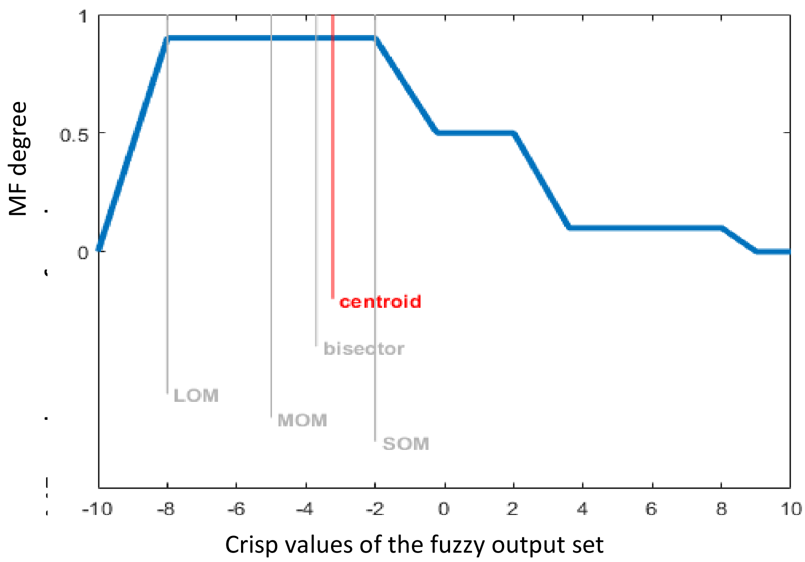


Figure 3.8 Defuzzification methods [166]

3.5. Designing a fuzzy logic decision-making scheme

In this section, the proposed fuzzy logic decision-making scheme for selecting the optimal sensing method for an application under the QoS awareness is described. The basic design of the proposed solution to select the proper sensing method based on the input parameters is shown in Figure 3.9.

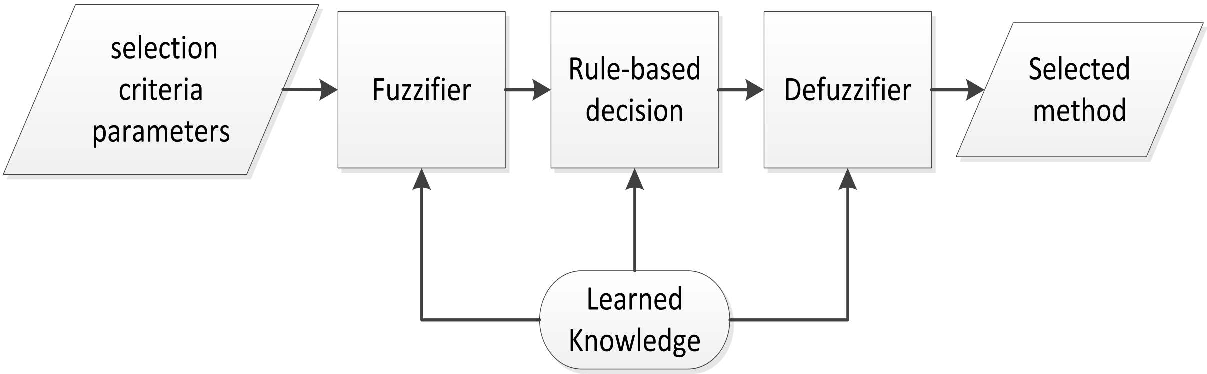


Figure 3.9 Fuzzy logic decision-making scheme for selecting the spectrum sensing method

First, the input parameters are fuzzified from measurable values to fuzzy linguistic variables to form the input membership functions (MFs). Based on the learned knowledge about the available sensing methods and operation requirements, a fuzzy rule-based decision-making scheme is designed, based on If-then statements, to map the input variables to the possible output of the scheme. The selection criteria input parameters, and defining the If-then rules for the rule-based decision, are explained in the following subsections.

3.5.1. Defining the selection criteria (FIS input variables)

The factors discussed in Section 3.3 are considered when defining the input selection criteria. The nature of these factors makes it hard to assign a precise or crisp value that measures them, but the fuzzy concept handles such parameters. To maintain the simplicity and efficiency of the proposed fuzzy system, the number of input variables should be minimised and their classification should be able to handle various essential changes. The input values must be strongly relevant and can be obtained, in most cases, during the system operation. In this study, taking into account that the proposed FIS will be implemented in a CR device to select the appropriate local sensing method, four input variables are defined that can cover the five aspects discussed in Section 3.3. In the subsections below, the four input selection criteria are described and classified.

3.5.1.1. Applications' QoS requirements (application QoS)

There are no widely accepted standard measurements of the QoS requirements of possible applications in CR. In the previous chapter, the correlation between sensing parameters and their impact on application QoS was discussed, and several simulations were conducted to study this correlation for various QoS applications and sensing parameter adjustments. The results of these simulations are reported in Chapter 5. The main QoS measurements directly affected by the sensing operation are delay, delay jitter, and throughput. There are numerous applications with different QoS requirements. If each of the specific application requirements is considered separately, this will produce a large set of inputs and lack of flexibility. Therefore, in this study the applications are categorised in terms of their QoS requirements, into four classes, as discussed in Section 2.4. The QoS requirements are classified into four levels, shown in Table 3.1. Any application can be classified into one of these levels, Very High, High, Medium (Med), and Low. These levels are the four linguistic fuzzy elements to be used for the required QoS level of the input variable, called Application QoS.

Table 3.1 Application classification based on QoS requirements

Application Class (Examples)	Sensitive to:			QoS level
	Delay	Jitter	Throughput	
Application A **Video conferencing**	High	High	High	Very High
Application B **Voice conversation**	High	High	Low	High
Application C **Email and Internet browsing**	Low	N/A	High	Med
Application D **Internet relay chat**	Med	N/A	Low	Low

Applications that are sensitive to delay, jitter and throughput belong to the 'Very High' level. 'High' is for applications that are sensitive to delay or jitter, but have low sensitivity to throughput. Applications sensitive to throughput and not very sensitive to delay are at 'Med' level. The 'Low' level is for applications with moderate sensitivity to delay and little sensitivity to throughput variations. The MF function for the ApplicationQoS input variable is shown in Figure 3.10.

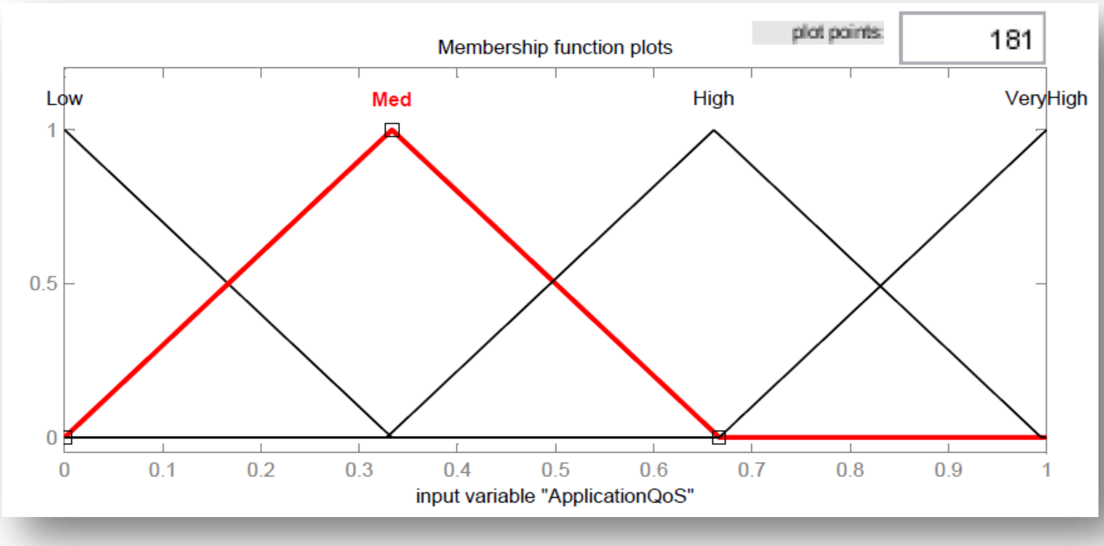


Figure 3.10 MF of the Application QoS input variable

3.5.1.2. CR capability

The CR capability input variable refers to the extent of the available resources for CR operation; in particular, the hardware resources of the CR device, as discussed in Section 3.3.4. A CR device capability is an important factor that must be included in the proposed decision-making scheme. For simplicity, the CR device capabilities are classified at three levels:

- 'High' for devices with adequate capability to run any sensing method;
- 'Med' for CR devices that have a moderate capability to run complex sensing methods but do not require sophisticated hardware;
- 'Low' for CR devices with limited capability, but enough to run simple sensing methods.

The MF function for the CR capability input variable, CRCapability, is shown in Figure 3.11.

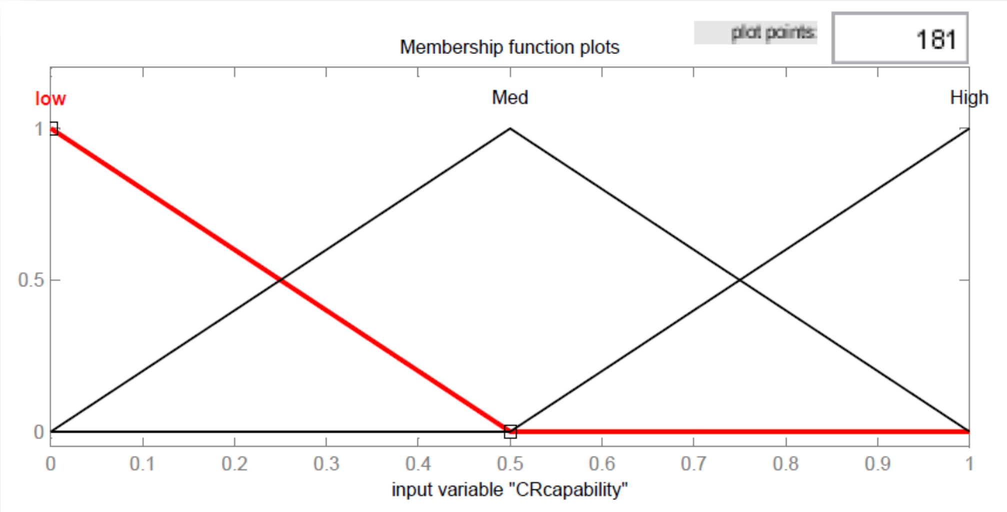


Figure 3.11 MF of the CR capability input variable

3.5.1.3. Prior information

The amount and type of information available about the PU signal are important factors when selecting the sensing methods. A CR device has to use a blind sensing method when there is no prior information about the PU signal, but when adequate information about the PU signal of a channel is available, it can choose among a wider range of sensing techniques, along with other factors, to sense the channel. This factor is also classified into three levels:

- ‘High’ for full information about the PU signal;
- ‘Med’ for sufficient information about the PU signal, for use by methods that are based on prior information;
- ‘Low’ for inadequate or no information about the PU signal.

The MF of the prior information input variable, Priorinformation, is illustrated in Figure 3.12.

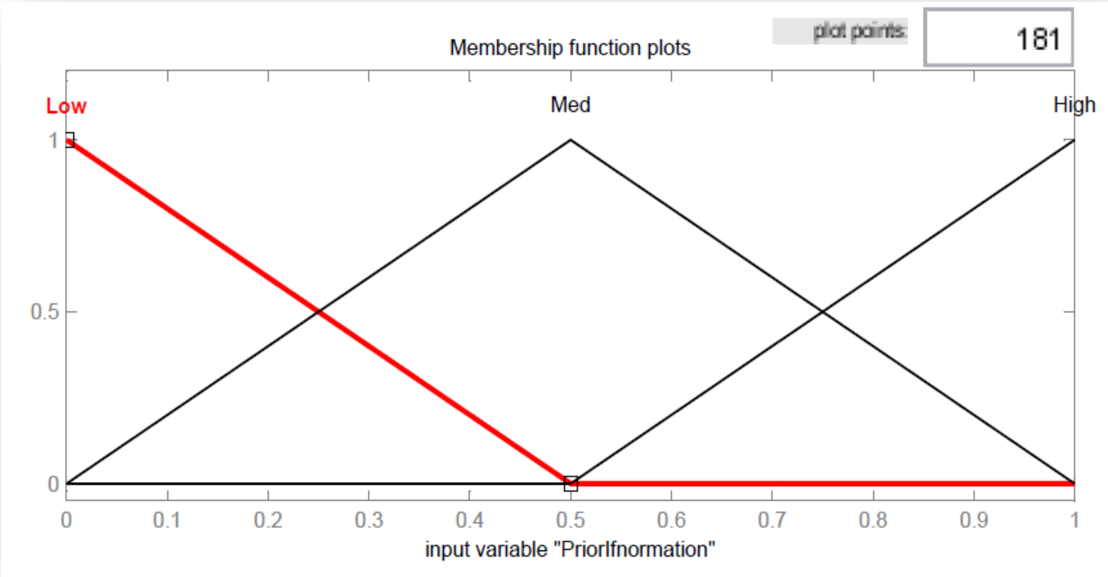


Figure 3.12 MF of the Prior Information input variable

3.5.1.4. The required PU protection

AS noted in 3.3.3, the required protection level for PU signals is an essential requirement. The PU protection input variable is used to consider this factor. The minimum level required of such protection is set according to IEEE standards, to 0.9 probability of detection for used the sensing method [155]. Thus, the minimum input of the PU protection variable is assumed to satisfy this requirement and the range of possible inputs for this factor reflects to what extent the protection should be increased, starting from the minimum requirement. Two levels of protection are adequate for this range:

- High for sensitive signals or applications; i.e., for PUs that demand higher protection than the minimum protection requirement.
- Low for non-sensitive signals and applications; i.e., for PUs requiring the minimum protection level.

The MF of the PU protection variable, PUProtection, is shown in Figure 3.13.

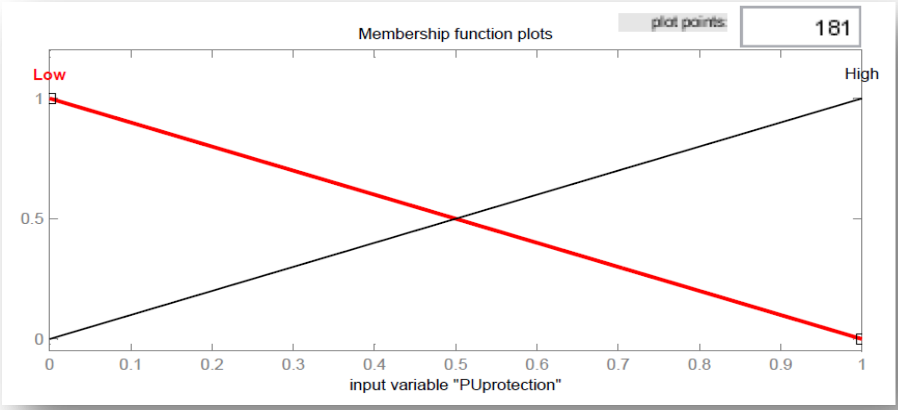


Figure 3.13 MF of the PU Protection input variable

3.5.2. Classifying sensing methods and defining FIS output variables

The best-known sensing methods were described and analysed in the previous chapter, and a summary of the comparisons between various sensing techniques presented in Figure 2.1 and Table 2.2. The classification of available sensing methods assists with the design of a fuzzy scheme for selecting an appropriate sensing method and to limit the number of output options to a reasonable number; it should offer various options including the best of each set of convergent characteristics. Accordingly, the sensing methods are classified into four main classes labelled Method 1, Method 2, Method 3 and Method 4, based on five parameters, as shown in Table 3.2. An example of a well-known sensing method in each class is shown in the last column of the table. The CR device should be reconfigured in real time with the best method available from any of these categories. The MF of the fuzzy output set is illustrated in Figure 3.14.

The cooperative sensing method is considered an additional stage, coming after the selection of the proper local sensing method. The cooperation is either enforced by a base station in a centralised system or conducted as an option for CR users in a distributed architecture. In both cases, CR users will use one of the local sensing methods and then share their information about the spectrum. The local sensing parameters may be reconfigured, based on the cooperative design, to provide a new overall sensing performance. Any cooperative sensing, based on its expected characteristics, may fall under one of the four method classes, but it will specify which local sensing technique is to be used for cooperation. Other possible considerations of cooperation could involve this feature in one of the selection criteria. For instance, it could be involved in the CR capability factor, discussed in Section 3.6.

Table 3.2 Classification of sensing methods for the decision-making scheme.

Method class	Sensing time	Robustness against SNR uncertainty	Performance	Complexity	Prior information required	Example
Method 1	High	High	High	High	High	Matched filter
Method 2	High	High	Med	Med	Med	Correlation
Method 3	Med	Med	Low	Med	Low	Covariance
Method 4	Low	Low	Low	Low	Low	Energy detection

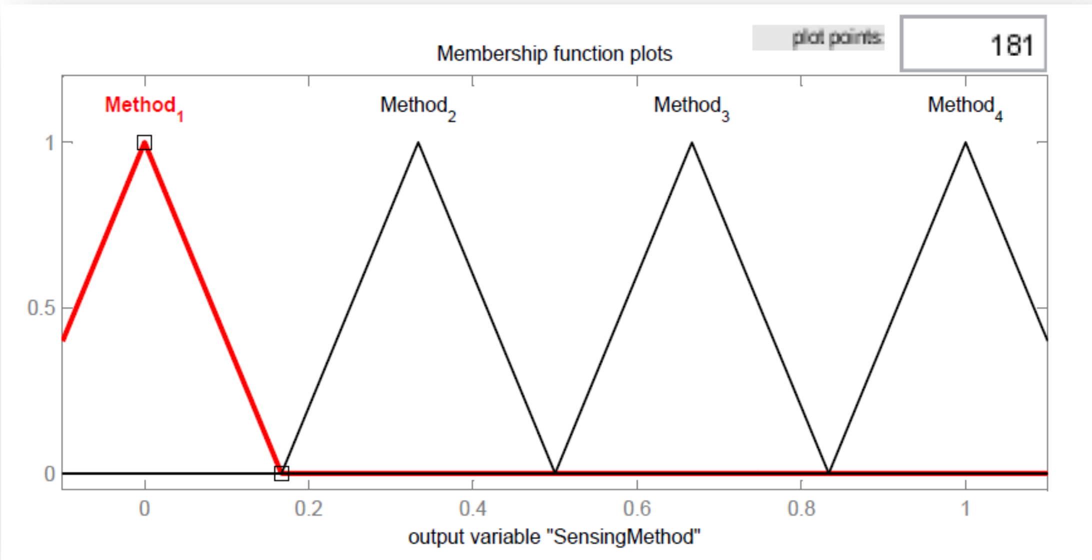


Figure 3.14 The MF of the FIS output set

3.5.3. Defining the If-then rules for the proposed FIS

The fuzzy logic decision-making scheme has to select the proper sensing method from four classes defined in the output set. The decision is based on the current situation presented by the input values and the pre-defined If-then rules of the proposed FIS. The input and output variables of the FIS have been discussed in Section 3.5.1 and Section 3.5.2 In this section, the rules are defined considering the following guidelines and assumptions:

- Blind sensing methods cannot distinguish PU signals from SU signals such as ED.
- Sensing methods based on prior information about PU signals, such as matched filter, can distinguish PU signals from other signals but require much longer sensing duration than blind sensing methods.
- Higher sensing performance, such as higher detection probability and lower false alarm probability, will provide higher PU protection and more likely to reach higher throughput.
- Methods with longer sensing times produce longer delays.
- Methods of a higher complexity require a higher CR device capability.
- When the CR device has inadequate prior information about the PU signal, only one of the blind sensing methods; i.e., Method 3 or Method 4 (see Table 3.2) can be used.
- When the CR device has a limited hardware capability, only a simple sensing method; i.e., Method 3 or Method 4 (see Table 3.2) can be used.

The rules are defined in Table 3.3 with the possible input MFs and the corresponding output MFs. The table shows the expected proper class of sensing methods for the given combinations of input parameters. Each row includes at least one rule. For instance, four rules, from 3 to 6, are included in the row starting with 3–6. For each row, when more than one value for an input parameter is present, the first value is for the first rules and the second value is for the following rules, and so on. The '**and(min)**' operation is used for combining the antecedent variables for all rules.

Table 3.3 Rule-Base for the Selection of the Spectrum Sensing Method

Rule	If Input (Combination operation for antecedent variables of is and):				Then Output
	Application QoS	CR capability	Prior information	PU protection	Method class
1	Very High	High	High	High	Method 2
2	Very High	High	High	Low	Method 4
3-6	Very High	High, Med	Med	High, Low	Method 2
7- 10	Very High	High, Med	Low	High, Low	Method 3
11, 12	Very High	Med	High	High, Low	Method 2
13- 15	Very High	Low	High, Med, Low	High	Method 3
16-18	Very High	Low	High, Med, Low	Low	Method 4
19	High	High	High	High	Method 2
20	High	High	High	Low	Method 3
21-28	High	High, Med	Med, Low	High, Low	Method 3
29,30	High	Med	High	High, Low	Method 3
31-33	High	Low	High, Med, Low	High	Method 3
34-36	High	Low	High, Med, Low	Low	Method 4
37, 40	Med	High	High, Med	High, Low	Method 2
41,42	Med	High	Low	High, Low	Method 3
43-46	Med	Med	High, Med	High, Low	Method 2
47,48	Med	Med	Low	High, Low	Method 3
49-51	Med	Low	High, Med, Low	High	Method 3
52-54	Med	Low	High, Med, Low	Low	Method 4
55-58	Low	High	High, Med	High, Low	Method 1
59-62	Low	Med	High, Med	High, Low	Method 2
63-65	Low	High, Med, Low	Low	High	Method 3
66-68	Low	High, Med, Low	Low	Low	Method 4
69,70	Low	Low	High, Med	High	Method 3
71-73	Low	Low	High, Med, Low	Low	Method 4

3.5.4. *Implementing the fuzzy logic decision-making scheme*

In this study, the logic designer toolbox in MATLAB R2014a is used to implement the proposed fuzzy logic decision-making scheme. The triangular MFs are used for the input and output variables, as shown in Figure 3.10 to Figure 3.13 for inputs and Figure 3.14 for the output. The Mamdani type is used to implement the FIS as it is more suitable for the proposed selection scheme than the Sugeno type. In addition to simplicity, the Mamdani type does not present output as linear or crisp. This is an advantage in the scheme being designed here, as the output is a sensing method, a fuzzy linguistic output, to indicate which sensing technique to use in the CR device. The '**and(min)'** operator is used for antecedents combination of all rules and '**min'** operator for combining the inputs in the antecedent of each rule. For output aggregation, the '**max'** operator is used and the **centroid** defuzzifyies the aggregated output. Most of the used default settings for the Mamdani type in MATLAB are shown in Figure 3.15. More details and snapshots of implementation of the proposed FIS in MATLAB are found in Appendix A.

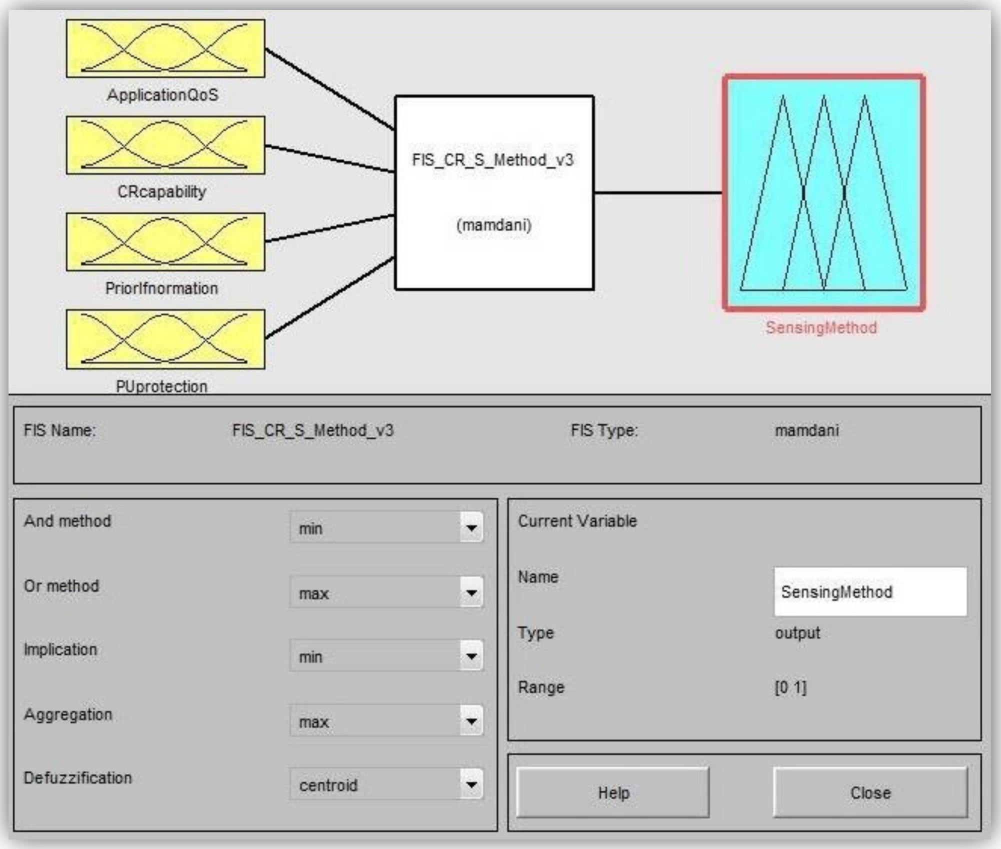


Figure 3.15 The proposed FIS implementation settings

The surface plot of the proposed FIS output for all possible values of the CR Capability and Application QoS input variables, when the other input parameters are fixed to 0.5, is shown in Figure 3.16. The output surface plot for the input variables Prior Information and Application QoS when other inputs are fixed to 0.5 is shown in Figure 3.17. The output surface plot for PU Protection and Application QoS variables when other inputs are fixed to 0.5 is presented in Figure 3.18.

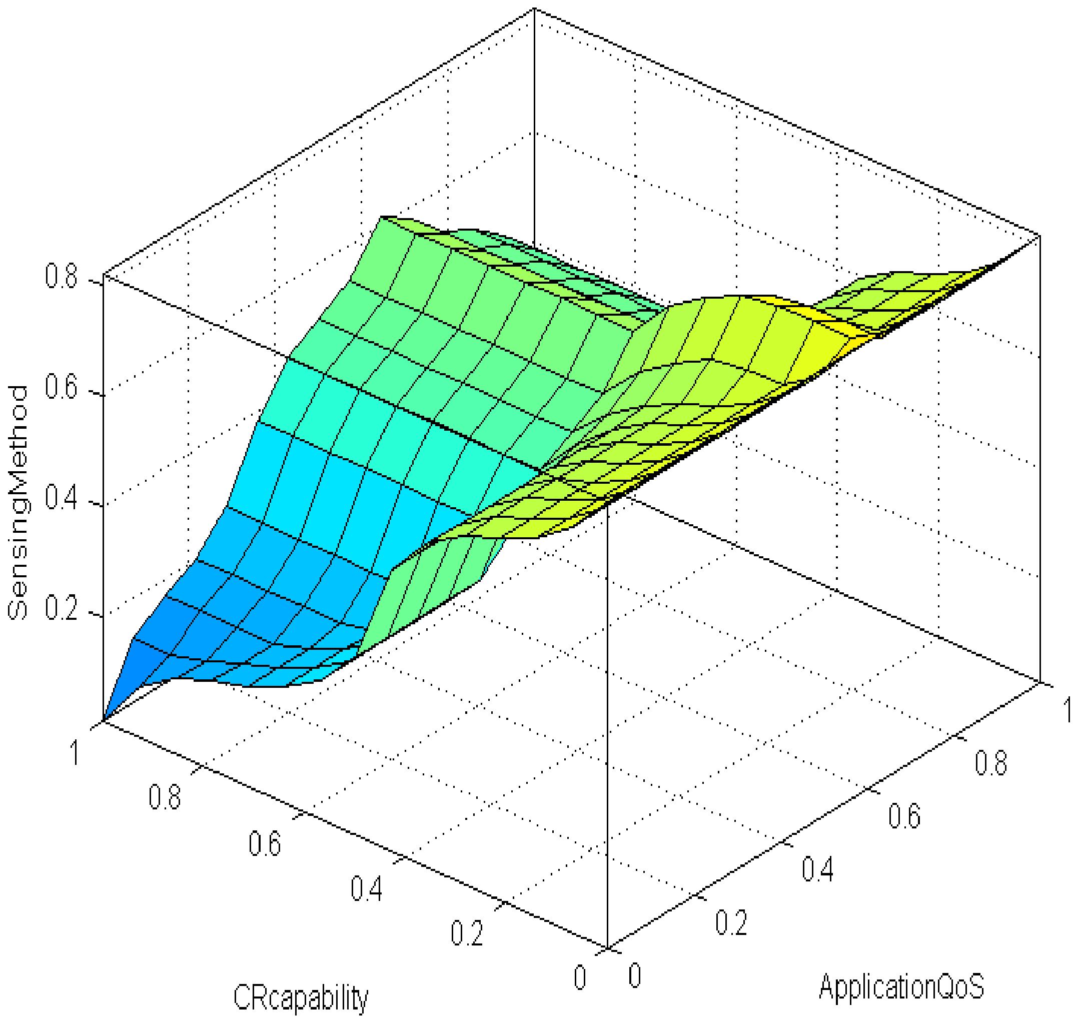


Figure 3.16 Surface plot of the input variables application QoS and CR capability (other input variables = 0.5)

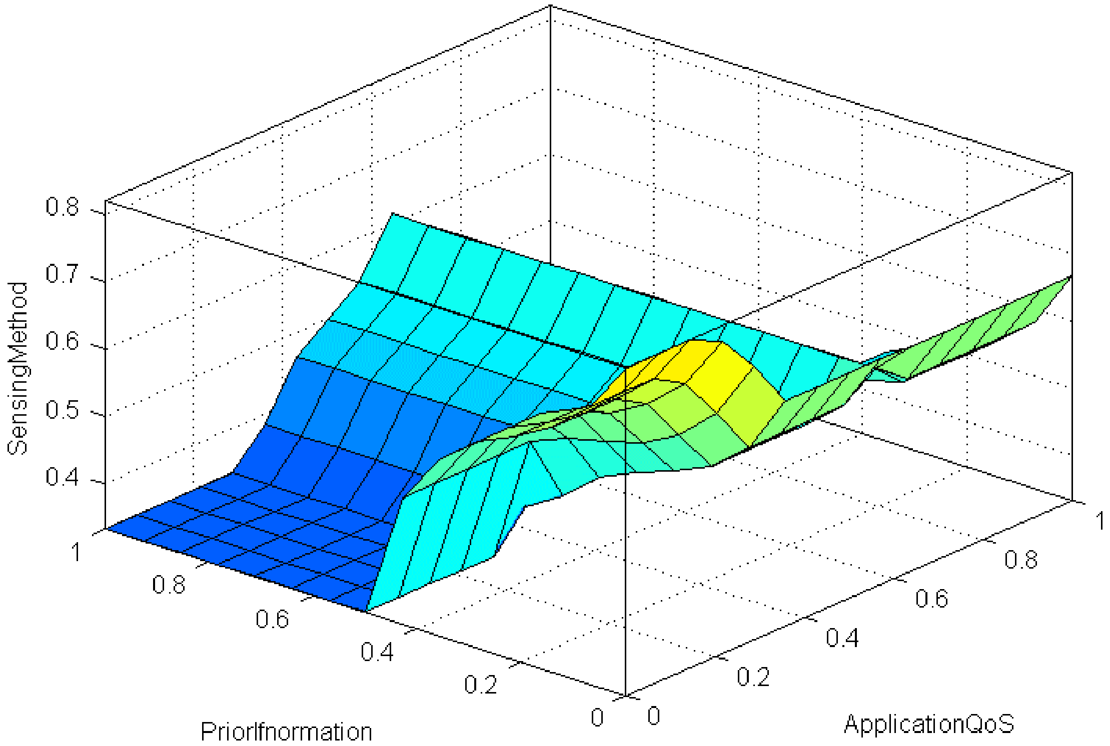


Figure 3.17 Surface plot of the input variables application QoS and Prior Information (other input variables = 0.5)

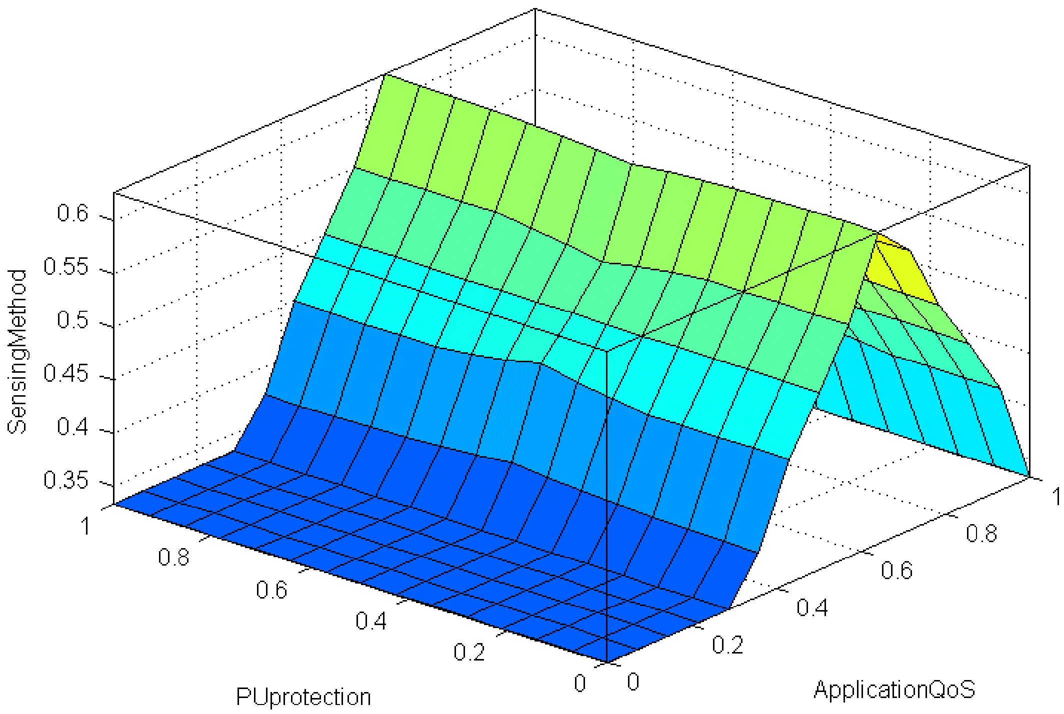


Figure 3.18 Surface plot of the input variables application QoS and PU protection (other input variables = 0.5)

The Application QoS variable is included in the illustrated plots for two reasons. First, the key goal of the scheme proposed here is to improve the overall QoS in CR. Second, the Application QoS variable is expected to contain the most dynamic value among the other input variables during a CR operation. The output shows that the CR device will adaptively use different sensing methods based on the input parameters. Figure 3.16 shows that the CR capability variable constrains the selection of the available sensing methods, i.e., either Method 3 or Method 4 (where values in vertical axis are more than 0.5), when the variable is less than 0.5. As the CR variable is increased, more different sensing methods can be selected according to changes in the Application QoS variable. Similarly, Figure 3.17 demonstrates how the Prior Information variable with values less than 0.5 limits the sensing selection to Method 3 or Method 4 when the QoS requirements, i.e., the Application QoS variable, are changed. For Prior Information variable higher than 0.5, other sensing methods can be selected for diverse Application QoS variable. However, Method 4 is excluded in this range as the PU protection variable is fixed to 0.5. An input variable may not affect the output selection when some other variables are fixed. For instance, Figure 3.18 shows that the PU protection variable has no influence on selecting the sensing output, as the surface plot is not changing along the PUprotection axis, besides the Application QoS variable. The PU protection variable will play a role in the output when the other inputs are changed to values higher than 0.5.

The MF of the output set (see Figure 3.14) is designed to have no overlapping elements; i.e., in the classes of sensing methods, so that the system can easily identify the selected method. After fuzzification, the resulting value is used to find which class to select. For example, if the defuzzification output is 0.125, as Output 1 (see Figure 3.19), then Method 1 is selected. In a case where the defuzzified output value falls between two methods, the CR node chooses the next method in the output MF scale. For example, if the defuzzified output is 0.5, as shown in Figure 3.19, Method 3 will be chosen as it comes after this value in the output MF scale. The input variables can be modified when implementing the MFs of the input factors based on a further assessment by the designer. They can also be adjusted by CR users based on their requirements. For example, the CR capabilities can be used to set different power saving themes: Low for power saving at the cost of performance, Med or High for better performance, utilising more power.

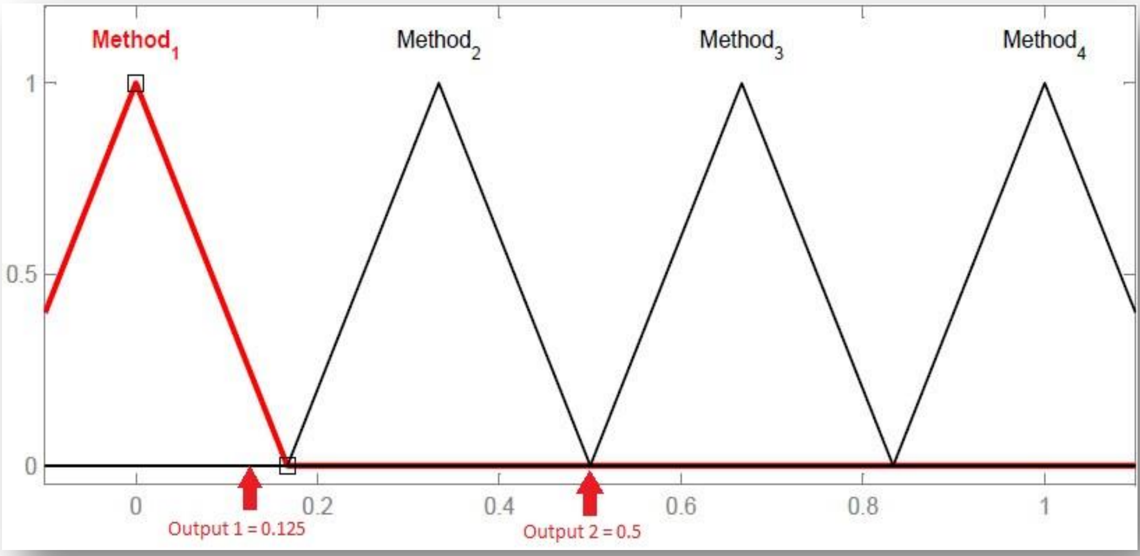


Figure 3.19 Selecting the sensing method based on the output defuzzification value

Information about the protection level required for a PU can also be estimated, based on available information about the PU signals or frequency bands of operation. For example, assuming that the PU of analog TV signals has low protection requirements while the digital TV signal user requires higher protection levels, a CR user can deduce the required protection levels of PUs by sensing the frequency bands of operation and relating them to the known bands for analog or digital TV broadcasts. The initial proposed rules are simply to illustrate the scheme, and are open to evaluation and modification based on further study and testing.

The specific sensing method can be pre-set in the scheme based on the best available techniques of that class. More sensing techniques can be used by dividing a class into ranges where each range is assigned to a different sensing technique. The implementation demonstrates that the scheme can be used in real time as it requires very low time and computation. It selects the proper method in one round and does not iterate; and it uses that method until the next period of sensing, when the selection may change depending on any variations in the input variables. Hence, a CR device in this proposed scheme can improve its performance, while a conventional device uses a fixed sensing method. For example, if the CR uses only ED, this will not provide high protection to a PU user when required; nor will it make optimal utilisation of the spectrum. However, when an input parameter has a value between two fuzzy values such as high and low, then more investigation is required to find the best shape used for each membership.

3.6. Summary and Discussion

As discussed in the previous chapter, using one sensing technique is not sufficient to achieve an optimised sensing operation, so the operations of CR and future networks need to utilise more than one spectrum sensing method. Each method and its operations affect the applications, which normally have varying QoS requirements, in different ways. The correlation between QoS and the sensing parameters has to be considered to achieve better QoS, besides maintaining adequate spectrum utilisation. This study comes to the conclusion that a CR device should select the most appropriate sensing method, based on the desired QoS among other requirements and constraints. In this chapter, five essential factors that influence such an approach to selection are pointed out and discussed. The application QoS level required for the current transmission and the required level of protection for PU signals in the operational band are two of these. The available information about the PU signals of the sensing channel, and CR device and network capabilities, act as selection constraints. Accordingly, a novel real-time fuzzy logic decision-making scheme that allows a CR device to select the appropriate sensing method for the current requirements and constraints is proposed. An important feature of this proposed scheme is that its time and computation overheads on CR devices are minimal. Moreover, the proposed fuzzy rule-based approach and the MF concept provide the system with the flexibility to update and improve its selection mechanism.

The proposed scheme of FIS with four input variables and one output variable is based on four classes of sensing methods. The input variables represent the selection criteria and cover most of the key factors that influence the selection of the proper sensing technique during CR node operation. The input variables are ‘application QoS’, ‘CR capability’, ‘prior information’ and ‘PU protection’. Limiting the number to four for input and output simplifies the system’s design and implementation without losing its effectiveness. The classification of the input and output and mapping rule can handle a variety of inputs and outputs. For example, the CR capability input variable can be set to represent any combination of selection aspects that limit the CR’s capability to run one or more sensing techniques. These aspects may include hardware limitations of the CR device, the available power resource, or the time to run sensing methods. The CR network mode and capability, discussed in Section 3.3.5, can also be considered in this input variable when cooperative sensing is one of the output selection options. However, cooperative sensing, as discussed in the previous chapter in Section 2.2.3, is based on one of the sensing techniques but adds a level of abstraction by exchanging or collecting the sensing outcomes of more than one CR node to come up with an improved decision. In such a case the CR network mode and capability should be considered in how the sensing operation is integrated with the MAC protocol used to access and share the communication medium. Also, the time available for sensing depends on the MAC protocol used, and such factors should be considered when designing a sensing strategy that includes defining when to sense, how frequently to sense, and whether the observed information about the RFS will be shared. In particular, the proposed decision-making scheme in this chapter is for local use by CR devices, to select the proper technique to stimuli the surrounding RFS and can be implemented in different sensing strategies. In contrast, the sensing strategy will decide when to use this decision-making scheme and how frequently, based on modes and capabilities that differ from one network to another.

Chapter 4. Sensing Selection Strategy for White-Fi

In this chapter the proposed sensing selection mechanism using a fuzzy logic based decision-making algorithm is applied to a promising CR network called White-Fi, which is expected to enhance existing and widely deployed Wi-Fi networks in terms of coverage and spectrum utilisation. This chapter demonstrates how the proposed selection mechanism may be customised to apply to White-Fi technology, resulting in a novel QoS awareness sensing strategy. An introductory background about White-Fi is presented in Section 4.1. Implementation issues concerning sensing in White-Fi are discussed in Section 4.2. The operational parameters of White-Fi that affect sensing and QoS are identified in Section 4.3. The proposed QoS awareness sensing strategy for White-Fi is described in Section 4.4. A summary and discussion are found in Section 4.5.

4.1. Introduction

A small number of frequency bands are unlicensed, including the industrial, scientific, and medical (ISM) bands which are used by a variety of indoor and short-range wireless communication systems such as Wi-Fi, Bluetooth, and Zigbee. These free unlicensed bands are not sufficient to handle the rapidly growing number of wireless devices using them, and modern applications running on these devices demand more bandwidth. These modern applications usually involve multimedia communications, such as media streaming, video conferencing and interactive gaming. With cognitive radio (CR) capability, wireless devices can operate opportunistically as secondary users (SUs) in licensed bands when they are unused, exploiting white spaces or spectrum holes. The spectrum holes of television bands are the most attractive for such opportunistic use of the spectrum because TV bands show high availability of white spaces and their use by primary users (PUs) can be reliably predicted as they have fixed operation schedules which can be obtained through Geolocation database (GDB) services [34]. Moreover, the TV spectrum is located below 1 GHz. Compared to the higher ISM bands, these frequencies offer more desirable propagation characteristics. For instance, they are lower and experience less attenuation.

An IEEE-802.11 protocol with CR capability is often referred to as CR Wi-Fi, White-Fi, Wi-Fi Like or IEEE-802.11af. The IEEE 802.11af, based on IEEE 802.11, is the first draft of the standard for CR networks operating in TV white space [65]. White-Fi devices are able to operate either in ISM or TV white space. The number of available TV white space channels and their frequency bands are regulated by governments. For example, in the United States, Federal Communications Commission (FCC) regulations allow the opportunistic use of frequency bands 512-608 MHz (16 TV channels, 21–36) and 614-698 MHz (14 TV channels, 38–51) [61]. Researchers should consider that White-Fi devices should be able to operate in white spaces in other licensed bands in the future. Figure 4.1 shows the main approaches to assessing the potential operating bands and their main conditions and actions. For ISM bands, only sensing is used where there are no PU signals. For TV white space bands, the current GDB approach is mainly to avoid interference with PU transmissions and sensing is used to avoid interference with other SUs. The potential white spaces in other frequency bands are not yet considered in the IEEE standard. GDB may not be possible in other frequency bands, so spectrum sensing is one of the most important functions in CR, used to identify available spectrum holes and to protect the PU from interference in any frequency band. However, sensing has an impact on the QoS that can be achieved by CR devices.

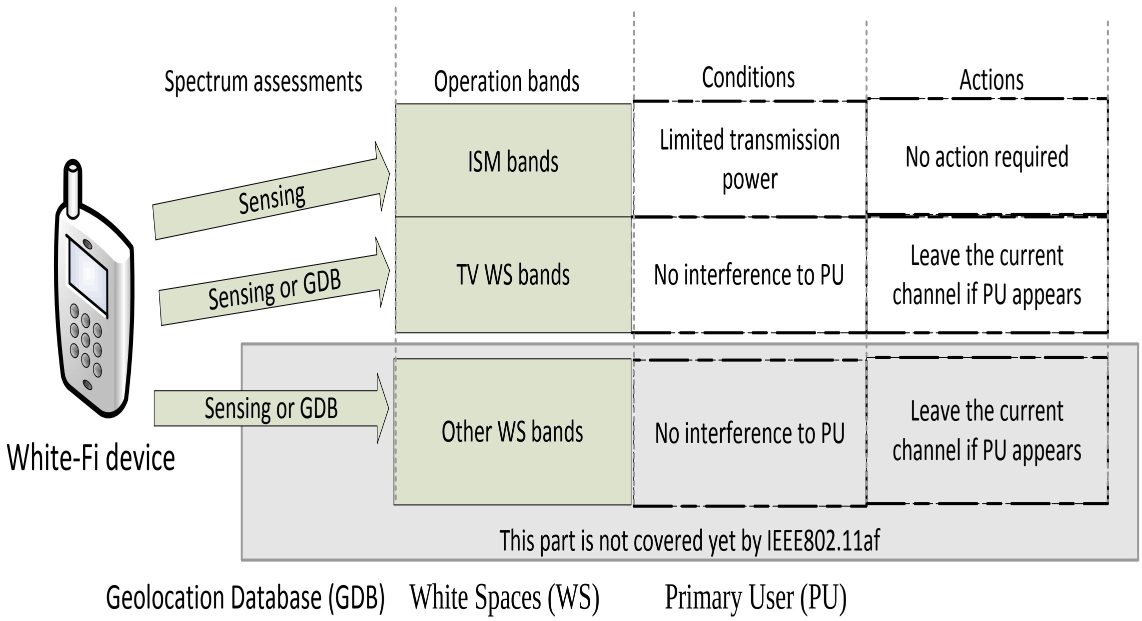


Figure 4.1 White-Fi potential operation bands and their required conditions and actions

Primarily the CR capabilities and requirements are established at the physical and medium access control (MAC) layers of wireless systems. This raises three fundamental questions: when to sense? How long to sense? Which channels to sense? Addressing these problems will differ from one SU network to another based on the approach used for accessing and sharing the communication medium. The carrier sense multiple access with collision avoidance (CSMA/CA) mechanism is used in wireless devices based on various 802.11 standards to share ISM bands [168]. The idea is that a Wi-Fi device checks a channel's occupancy before transmitting over it. Typically this is done using a simple sensing technique like energy detection (ED), where the energy of the channel is measured and compared to a predefined threshold. If the measured energy level exceeds the threshold, the channel is in use by another device. Otherwise, it is idle (see Section 2.2.1.1 for details). A Wi-Fi channel can be used only if it is inactive, and the transmission will be deferred for an arbitrary time if the channel is found to be occupied.

In the case of White-Fi, SUs have to compete for the same available white spaces without interfering with the PUs of these bands. Therefore, the CSMA/CA used in Wi-Fi needs to be modified to handle the new requirements of White-Fi operations in white space bands if spectrum sensing is to be used to identify white space bands. The sensing function, particularly when there is no GDB, has an important role among other CR functions in White-Fi environments. White-Fi is mainly proposed to use the available white spaces under an essential restriction where the PU must be protected and has the highest priority in using its license band. The white spaces can be shared by different wireless standards and technologies as long as the devices comply with regulations. Accurate spectrum sensing is needed to protect PUs efficiently and to share the white spaces with other SUs. However, using high-accuracy sensing methods can result in a significant negative impact on upper layers, in particular, the application layer.

4.2. Sensing function implementation issues

Sensing is needed in White-Fi for two main reasons: first to avoid interference with other White-Fi users; and second, and most importantly, to prevent interference with PUs. Implementing CR functions in general and sensing functions, in particular, may cause issues in White-Fi environments. The ISM channels are openly shared by wireless devices using different standards and technologies as long as they comply with some transmission regulations. This situation applies to the available white spaces, with the additional restriction that the PU must be protected and has first priority in using its licensed band. Sensing, in White-Fi, has to be efficient and accurate to satisfy these requirements. This section points out some of the implementation issues of the sensing function for White-Fi environments.

4.2.1. Sensing the primary user

In CR technology, to protect the PU, an SU will transmit on the PU's channel only when it is not used by the PU, and will stop transmitting when the PU appears. The SU can use the GDB to obtain a list of the available channels in its location; the list is valid for a specific period. This means an SU has to obtain updates from the GDB periodically, or obtain them immediately if the SU moves to new location. This approach is proposed for analog TV white space bands, where it is possible to gather the operating schedules of the different PUs of these bands and provide them to SUs in a GDB. The database has to be accessed using a supplementary communication method, such as via satellite, by both the GDB server and the SUs. This approach cannot be used by all potential SUs as not all are capable of using diverse communication technologies and accessing various channels simultaneously. Also, a PU may not have a fixed schedule on which to use its channel, or may be unwilling to provide this information to the GDB. In such cases, spectrum sensing becomes the most essential and usable approach for detecting the PU.

Sensing methods that do not require prior information about the PU signal, such as ED and covariance-based detection, can be used to detect the presence of signals from other users on the sensed channel, but they cannot tell if a signal belongs to a PU or another CR user. For more accurate recognition of a PU signal, prior knowledge about the signal is required. The matched filter sensing method is a well-known method, based on complete prior knowledge about the PU signal. To detect the presence of a PU from its signal accurately, the matched filter sensing method requires a dedicated receiver for each different possible transmission technology used by PUs [169]. In general, the sensing accuracy is a function of the probability of positive detection (P_d) and the probability of false detection (P_f) under the given operating conditions. False detection is sometimes referred to as false alarm detection. Typically, achieving higher sensing accuracy requires more overhead regarding the sensing duration, sensing frequency and computation. The sensing overhead has its impact on QoS metrics, such as the throughput, delay and jitter, of an SU. A trade-off between the sensing accuracy and the various overheads is required for better performance [30]. In the literature, P_d and P_f are usually determined under the assumption that all detected signals are PU signals, while in reality, some can be from other SUs. Moreover, modelling the PU traffic with the assumption of an ON/OFF operation may not reflect the wide range of potential PU traffic that follows different operational models ,such as in cellular and wireless sensor networks [60]. Taking into account the actual P_d and P_f, and unpredictable PU behaviour, is necessary if more accurate sensing of CR-based networks is to be achieved.

4.2.2. Sensing other secondary users

Different SUs can exist and operate within the same coverage area. In addition to protecting the PU, an SU within the same coverage area as other CR users has to detect the presence of other SUs and share the available white spaces with them. On a White-Fi network, there may be two types of user: pure White-Fi (or homogeneous) SUs, and heterogeneous SUs.

4.2.2.1. Pure White-Fi SUs

In this scenario, all SUs sharing the same channel are White-Fi users. A White-Fi user using a shared channel has to vacate the channel if the PU appears. Otherwise, the channel can be shared with other White-Fi users using a CSMA/CA mechanism. The main issue for White-Fi users sharing a channel is how to distinguish between the PU and other White-Fi users. To make this distinction, a longer sensing duration and a more complex sensing method need to be used. Moreover, complete knowledge of the PU signal is required if there is no GDB for that channel. Once the PU detection issue is addressed, White-Fi users can share the white space bands based on a CSMA/CA mechanism. Typically, the sensing used in CSMA/CA is not able to distinguish a PU signal from other Wi-Fi users' signals, so White-Fi users cannot efficiently use the available white spaces without interfering with their PU signals by using the existing technologies for Wi-Fi channel access control.

4.2.2.2. Heterogeneous SUs

In this scenario, White-Fi users and SUs based on other different wireless technologies exist in the same available white spaces. Under this scenario, a White-Fi user needs to distinguish between three types of user: the PU, other White-Fi users, and other non-White-Fi SUs. An example of such scenario is when White-Fi users and IEEE 802.22 users are sharing the same TV white spaces. While White-Fi users can back off when the white space channel is occupied by 802.22 transmissions, 802.22 users may not have the ability to sense White-Fi transmissions. This will have a negative impact on the performance of the SUs and their ability to share the network fairly, particularly, for those using the 802.22 standards, where a GDB is used to determine the TV white spaces [170]. IEEE 802.19.1-2014 enables the family of IEEE 802 wireless standards to coexist and effectively use TV white spaces based on the use of the GDB of the TV white space bands [171]. The proposed architecture and algorithms in IEEE 802.19.1-2014 are still an open area for new patents and research. When the GDB is not available, the different SUs sharing the same white spaces are solely reliant on sensing to distinguish between SU and PU signals. The coexistence of standards other than the IEEE 802 family is another challenge. The sensing function will be essential when different standards coexist in white space bands where the GDB is not available. The main obstacle to addressing coexistence, in this case, is how the sensing operation can effectively differentiate between the PU and other SUs of other standards. Although advanced sensing, such as matched filter sensing, may distinguish between signals when prior information about these signals is available, a higher sensing duration is required. Moreover, fairly sharing a white space between heterogeneous CR systems becomes very challenging.

4.2.3. Synchronisation and control channel with other nodes

In White-Fi the CR nodes, including the access point (AP), may change to another channel for receiving and transmitting data during operation, particularly once the PU is detected. The CR nodes that belong to the same network have to synchronise their data channel during the handoff to maintain their connections. This may require another channel, called the common control channel (CCC), rather than using the data channel. The CCC can be used to exchange control information among SUs that may be related to competing and accessing the data channel among other nodes, spectrum sensing, and management outcomes [172].

In centralised networks, the control and data channels are decided by a central base station or an AP, as on IEEE 802.22 and IEEE 802.11af networks. In conventional Wi-Fi, when a node loses the connection with the AP it will scan to find a new channel by finding the beacon frames of that AP. In the case of CR, there will be a larger number of possible channels and their bandwidth may vary, and scanning for a new channel to connect the AP to may take a substantial amount of time [173]. Furthermore, nodes at different locations may have different spectrum opportunities to access a channel when the PU is absent. For instance, there may be some parts of the spectrum which are only available for low-power CR signals for short-range or low-rate transmission, because of the locations of the nodes [174]. On networks operating in infrastructure mode, a central controller like AP or a base station helps in handling synchronisation between nodes of different transmission rates on the same network [174].

In ad-hoc mode, which is a decentralised architecture, synchronisation becomes a more complicated task and creates more overhead [175]. Another level of synchronisation could be required when various networks coexist in available white spaces. In such an environment, synchronisation among all the communicating systems and protocols is based on a universal reference clock and global CCC [170]. Many questions and challenges are raised on how to establish and allocate CCCs in CR networks. For instance, does the CCC have to be predefined or dynamically determined? How many CCCs are required, and how much bandwidth is needed for each channel? Should the CCC be created within the licensed, free or white space bands?

Additional spectrum channels are required, as at least one will be used for exchanging data and another for exchanging control messages. Consequently, a dedicated CCC among nodes in the same network, or a global dedicated CCC among different networks, imposes the need for more than one transceiver for each CR node. Dynamic allocation of a CCC requires precise time synchronisation and complex algorithms, particularly in the absence of a centralised regulator and of agreed standardisation [176]. Because of such limitations in using CCCs, another approach, based on avoiding the need for a CCC in CR operation, has been proposed [177]. In this case an efficient mechanism by MAC protocol to share the same channel for data and control information becomes essential.

In relating the CCC to the sensing function, there are two main issues to be considered. First, establishing a CCC requires sensing when it has to use the available white space bands. Second, decisions about channel access may rely on a CCC for exchanging spectrum information among SUs, particularly in cooperative sensing. Hence, when designing and implementing a sensing strategy, the availability of a CCC and synchronisation are important factors to be considered. For wide adoption of proposed CR solutions, especially for decentralised networks, the need for synchronisation and a CCC should be avoided. Also, the required spectrum resources and wireless interfaces should be reduced in CR devices.

4.2.4. Requirements for upper layers

Typically CR functions are implemented within the MAC and physical layers. However, upper layers are directly affected by the overheads created by CR functions at the MAC and physical layers. The cross-layered design is the main key to exploiting the best of each layer by exchanging information between them so they can adapt dynamically to the changing nature of the CR operating environment. Fundamentally, in the cross-layered design the parameters of the MAC and physical layers can be optimised for better overall performance of the communication layers. For example, the carrier sensing errors in CSMA/CA networks, measured by the miss detection and false alarm of sensing, and their direct effect on throughput and delay were formulated and analysed in [178], and expanded to CSMA/CA networks with CR capability in [179].

In general, the cross-layer approach in sensing aims to find an optimal trade-off between the required QoS and sensing accuracy [180]. Practically, the aim can be narrowed to improve at least one of the QoS metrics, such as delay or throughput, without compromising the level of protection required by the PU. In Wi-Fi, the TCP and routing protocols, used in the transport and network layers respectively, play an important role in QoS metrics [181]. This fact leads to new directions for research into improving the adaptation of the transport and routing protocols to the MAC and physical layers on CR networks, for better performance of CR devices. Any improvement in performance should require minimum modification in the upper layer protocols to maintain compatibility with existing protocols and systems. For example, the sensing outcome is sent directly to the transport layer, so frame size, spectrum access decision, modulation and coding schemes will be configured to help the TCP protocol achieve higher throughput [55]. In this thesis the focus is on the cross-layered design required between the application and MAC layers regarding spectrum sensing and QoS requirements.

4.3. White-Fi operation parameters affecting sensing and QoS

In this section, different factors and requirements that could be considered to mitigate the sensing function implications in White-Fi are studied. The main focus is on areas that are specific to White-Fi technology.

4.3.1. Application QoS requirements

A CR user may run different applications that may require diverse QoS requirements. For instance, real-time applications are more sensitive to delay and jitter than email applications. QoS requirements depend on which applications are running on the CR device, and may vary during operation. For better QoS, the sensing parameters should be adjusted accordingly in real time, and this imposes the need for a cross-layered design to handle the exchange of information between different layers [182]. The sensing operation has an impact on delay, achieved throughput, and packet loss. The main parameters of the sensing operation that affect QoS are the sensing duration, sensing frequency and sensing accuracy (P_d and P_f). These parameters can be adjusted to mitigate degradation in the QoS caused by having to perform sensing under the constraints of protecting the PU and achieving high spectrum use.

4.3.2. Operation adjustments

The 802.11-based networks have different operational adjustments that affect their performance. Different operational situations may require different settings of the operation parameters, and for improvement of performance, these settings should be adjusted intelligently and dynamically to adapt to real-time changes to the operational conditions. The size of the data frame, the use of control frames, and the frame classification for prioritisation to enhance QoS are the main operational adjustments covered in this section. The 802.11af inherits these possible adjustments from non-CR 802.11 standards, so these possible adjustments and their performance should be researched under the new operational requirements of 802.11af by considering the level of compliance to the general regulations by big standard organisations.

4.3.3. Fragmentation and aggregation

In Wi-Fi, packet fragmentation has been used, so that when a packet exceeds a given size, it is fragmented and each part is carried by a smaller frame at the MAC layer. The resulting MAC frames are transmitted independently and are to be acknowledged separately. As the wireless medium is error-prone, this mechanism was proposed to reduce the influence of noise errors on transmitted data and to improve reliability [183]. In CR networks, it also reduces the data transmission period and increases the sensing frequency, and consequently, fragmentation becomes more important in the case of CR operations. It helps to increase protection of the PU from an SU transmission, and to reduce the possibility of an SU's ongoing transmission interfering with the PU. Another important factor related to these benefits of fragmentation is the sensing operation. Sensing should be performed before any SU transmission to avoid collision with transmissions of other users of the channel. In CR networks, where the CR user is assigned a dedicated transmission channel based on frequency or time, sensing is usually conducted periodically in a fixed time interval, sequentially with the data transmission period. In random access networks, this approach cannot be conducted efficiently because the data transmission period is not fixed but depends on unpredictable channel conditions.

Accordingly, sensing frequency should be a function of the size of fragmentation. A smaller fragmentation size causes smaller transmission frames and more frequent sensing. In general, for the same sensing method, a longer sensing duration helps to reach a higher P_d and lower P_f under the same spectrum conditions. There is little in the literature on how to optimise the combination of sensing duration, fragmentation threshold and sensing frequency in random access networks that will improve overall QoS of running applications. Moreover, the use of various sensing methods, of different accuracy, has not yet been fully researched. In addition, the capability to distinguish a PU signal from other signals must be taken into account.

Instead of fragmenting a larger frame into smaller frames, a frame aggregation scheme is used to aggregate upper layer frames to form a larger frame at the MAC layer, like the aggregation MAC protocol data unit (A-MPDU) of IEEE 802.11n. This opposite approach to fragmentation is used when the transmission channel is large enough for bigger frames to be sent in a short time. The fragmentation approach offers more reliability at the cost of more protocol overhead for legacy 802.11, whereas the aggregation approach reduces the overhead for high-speed IEEE 802.11 networks [184]. For an aggregated MAC service data unit (A-MSDU), the aggregated frames belong to the same service category and have the same destination. The maximum A-MSDU length that an 802.11n base station can receive is 3839 octets or 7935 octets [185]. A-MPDU upper layer frames being sent to the same destination are aggregated to form a frame with a length of 8191, 16383, 32767 or 65 535 octets [185]. Several MSDUs can be aggregated in one larger MPDU, and then one or more MPDUs are encapsulated into a single physical service data unit (PSDU) [186]. Before transmitting the PSDU, the physical layer convergence procedure (PLCP) preamble and header are added to the PSDU to form the actual frame to be sent in the physical layer called PLCP Protocol Data Unit (PPDU). Similar aggregation schemes are used in 802.11ac but with slight differences in frame sizes. For instance, the maximum size of PSDU in 802.11n is 65534 octets but in 802.11ac is 1048575 octets [187]. The block-acknowledgement (B-ACK) frame is proposed to acknowledge the aggregated frames instead of using single ACK for each successfully received MPDU frame. The aggregation and B-ACK schemes help to reduce the overhead of exchanging control messages, and potentially achieve higher throughput. The final form and size of the physical frame is limited to the available channel bandwidth. For 802.11af, the possible aggregation settings depend on available white space bandwidths. For TV white space, the width (W) of the Basic Channel Unit (BCU) can be 6, 7 or 8 MHz, depending on the regulatory domain. The 802.11af standard defines different contiguous and non-contiguous bandwidths that can be formed from the available BCUs, for TV very high throughput (TVHT), as listed below [65]:

- 2W: two contiguous BCUs (12 MHz, 14 MHz, or 16 MHz)
- W+W: two non-contiguous BCU (6+6 MHz, 7+7 MHz, or 8+8 MHz)
- 4W: four contiguous BCUs (24 MHz, 28 MHz, or 32 MHz)
- 2W+2W: two non-contiguous frequency segments where each segment is composed of two BCUs (12+12 MHz, 14+14 MHz, or 16+16 MHz)

The standard sets the maximum PSDU to 1065600 octets when the highest bandwidth of 32

MHz (4W or 2W+2W) is used [155]. Sensing duration and frequency are affected by the fragmentation/aggregation settings because sensing cannot be conducted while a physical frame such as a PPDU is being transmitted. The maximum duration for transmitting a PPDU, aPPDUMaxTime, is set to 20 ms in 802.11af [155]. More study is required to find the optimum size of the PPDU based on factors such as sensing duration, available channel width and the desired PU protection.

4.3.4. RTS/CTS settings

The hidden node problem is common in wireless networks where at least two wireless nodes, called hidden nodes, cannot sense the presence of each other because of spatial or physical limitations. Collisions are likely to occur at the same destination of the hidden nodes transmissions if they are transmitting at the same time. The 'request to send/clear to send (RTS/CTS)' mechanism is used in IEEE 802.11 networks to avoid this problem. When this mechanism is used, the node wanting to transmit sends an RTS frame to the destination before sending the actual data frames, and will send a data frame only if the destination returns a CTS frame. RTS/CTS frames incur more overhead and hence lower throughput when there are no or only a small number of hidden nodes, therefore, enabling RTS/CTS improves network performance when there is a large number of potentially hidden nodes and the amount of data to be exchanged is also large [188]. The use of RTS/CTS frames is more efficient for large data frames in that there is no fragmentation exchanged in high-density IEEE802.11 networks [189].

To enable the RTS/CTS mechanism only when it is expected to reduce collisions, a threshold, called the RTS threshold, is used to enable the mechanism when the frame size to be sent exceeds the RTS threshold [190]. The RTS threshold plays an important role in the wireless network performance and there is no one optimised value that can be generalised and applied to different networks [191]. In White-Fi, the RTS/CTS mechanism will only work between SUs that can understand RTS/CTS frames and use the CSMA/CA mechanism. RTS/CTS are not suitable for solving hidden node problems if other SUs and PUs cannot understand them. In Wi-Fi, as channel sensing is conducted only before sending the RTS frame, a data frame will be sent after the CTS is received without sensing. During the exchange of RTS and CTS frames between source and destination, contending nodes remain idle, based on the information carried by the RTS and CTS frames. The idling includes the period required for exchanging the data frame. During this period, there is no need to sense the channel physically; so it is called a virtual carrier-sensing mechanism when it is used to avoid collisions.

The information carried by RTS, CTS and data frames includes the duration required for data exchange. Any node that is not participating in the ongoing transmission uses this information to update its time counter, the network allocation vector (NAV), which is used to defer a transmission until the end of the ongoing one. An alternative way to update the NAV is by sending a CTS-to-self control frame. A CTS-to-self control message is used when the use of RTS/CTS is not efficient, particularly in environments coexisting with legacy 802.11 technologies [192]. The use of CTS-to-self messages, in TV white space, seems unnecessary as no legacy 802.11 devices operate in that band. In White-Fi, the PU or other SUs using the same channel may not be able to understand control messages used for virtual carrier-sensing, so the conventional use of RTS/CTS and NAV mechanisms are not sufficient to avoid collision with PU transmissions. Moreover, when the number of SUs increases, the number of RTS-CTS messages and packet collisions will increase, with a further reduction in the network efficiency of the SUs [193]. In addition, during the time of transmitting an RTS and waiting for the CTS response, the transmitter cannot sense the transmission channel, unless an extra channel like CCC is used to exchange control messages or the sensing technique in use can identify the potential PU signal from other signals. Despite that, the RTS/CTS mechanism is still useful in ISM bands to reduce interference, particularly between hidden nodes. As the mechanism is already included in 802.11, it could be used to enhance the performance in White-Fi and to exchange information regarding CR operations as proposed in [194, 195]. In fact, the RTS/CTS mechanism was not originally proposed for CR operation and it is only an optional feature. More study is required to reduce its overhead while extending its benefits in the new opportunistic CR environment.

4.3.5. 802.11e mechanisms for QoS enhancement

The IEEE 802.11e standard is proposed to enhance QoS in IEEE 802.11 networks [196]. As applications have different requirements, in 802.11e the frames belonging to different applications are ranked with one of eight user priority levels. In contrast, previous IEEE

802.11 standards used either the distributed coordination function (DCF) or point coordination function (PCF) mechanisms at the MAC layer, where best-effort service is provided equally to all traffic streams from different applications accessing the medium. A new coordination function, the Hybrid Coordination Function (HCF), is introduced in IEEE 802.11e to enhance QoS in IEEE 802.11 networks. The HCF accommodates two medium access methods: a distributed contention-based channel access mechanism, the enhanced distributed channel access (EDCA), and a centralised polling-based channel access mechanism, the HCF controlled channel access (HCCA). The EDCA has become a mandatory feature in IEEE 802.11-based devices, including IEEE 802.11af. In contrast, the HCCA is optional and rarely seems to be implemented or used in commercial APs [197]. The EDCA is based on the basic DCF, which can be used on both ad-hoc and infrastructure modes. The HCCA is more complex and based on PCF, so it can only be used in an infrastructure mode where an AP is required for its centralised coordination mechanism. Hence, this study focuses on EDCA, as it is widely adopted and can be used for different network modes.

In IEEE 802.11e three parameters, the arbitration inter-frame space (AIFS), contention window (CW) size, and transmission opportunity (TXOP), are used for prioritising traffic streams. For EDCA, four access categories (AC): voice (AC_VO), video (AC_VI), best-efforts (AC_BE) and background (AC_BK), are defined. These categories are assigned different priorities, from highest to lowest respectively, and mapped to the eight user priority levels. The category AC_VO has top priority and is usually given to traffic carrying voice information. It is followed by the AC_VI category for video traffic and then the AC_BE category for data traffic. The category AC_BK has the lowest priority and is usually assigned to background traffic that is not carrying application layer data. The short inter-frame space (SIFS) value, aSIFSTime, is used as the shortest inter-frame space (IFS) value for transmitting high priority frames such as data acknowledgement and CTS frames. An example of how the channel access prioritisation mechanism works in EDCA is illustrated in Figure 4.2, which is inspired by work in [198]. A time slot, aSlotTime, in DCF is the basic time unit required for a sensing interval where typically ED is used to identify if the channel is busy or idle. This assessment, carried out in the time slot, is called clear channel assessment (CCA) or CCA-ED as the ED is usually used for its sensing (IEEE 802.11 2012 standard, section 9.3.7) [199]. Each AC has a contention window (CW) with a specified minimum and maximum size. For example, $CW_{min}[AC]$ and $CW_{max}[AC]$. An arbitration inter-frame space (AIFS) value and a TXOP interval are used to support the QoS prioritisation [200]. Instead of using a fixed distributed inter-frame space (DIFS), also called a DCF inter-frame space, the AIFS[*AC*] value is a variable value, calculated based on the AIFS Number (AIFSN) of the frame AC, AIFSN[*AC*], as shown below:

$$AIFS[AC] = AIFSN[AC] \times aSlotTime + aSIFSTime \tag{4.1}$$

An AIFS[*AC*] value determines the time that a node defers access to the channel when it is idle, based on the assigned AIFSN for that AC. Some default values of EDCA parameters proposed for IEEE 802.11af are summarised in Table 4.1 (derived from the standard amendment 5 [201]). For instance, possible values for AIFSN[*AC*] are seven for AIFS[AC_BK], three for AIFS[AC_BE], and two for AIFS[AC_VI] and AIFS[AC_VO] (see Table 4.1). For the frames belonging to the categories of AC_VI and AC_VO, the AIFSN[*AC*], $CW_{min}[AC]$ and $CW_{max}[AC]$ values are set smaller than for the frames of the categories AC_BE and AC_BK, to reduce delays. Therefore, higher priority frames access the operational channel earlier than other frames in the same transmission queue, as shown in Figure 4.2.

Table 4.1 EDCA default parameters

AC	CWmin[AC]	CWmax[AC]	AIFSN[AC]	TXOP limit for:	
				HT*, VHT**	TVHT
(AC_BK)	aCWmin = 15	aCWmax = 1024	7	0	0
(AC_BE)	aCWmin	aCWmax	3	0	0
(AC_VI)	(aCWmin+1)/2-1 = 7	aCWmin	2	3.008 ms	22.56 ms (for BCU 6 or 7 MHz) 16.92 ms (for BCU 8 MHz)
(AC_VO)	(aCWmin+1)/4-1 =3	(aCWmin+1)/2-1	2	1.504 ms	11.28 ms (for BCU 6 or 7 MHz) 8.46 ms (for BCU 8 MHz)

** High throughput (HT), e.g., 802.11n ** Very High Throughput (VHT), e.g., 802.11ac*

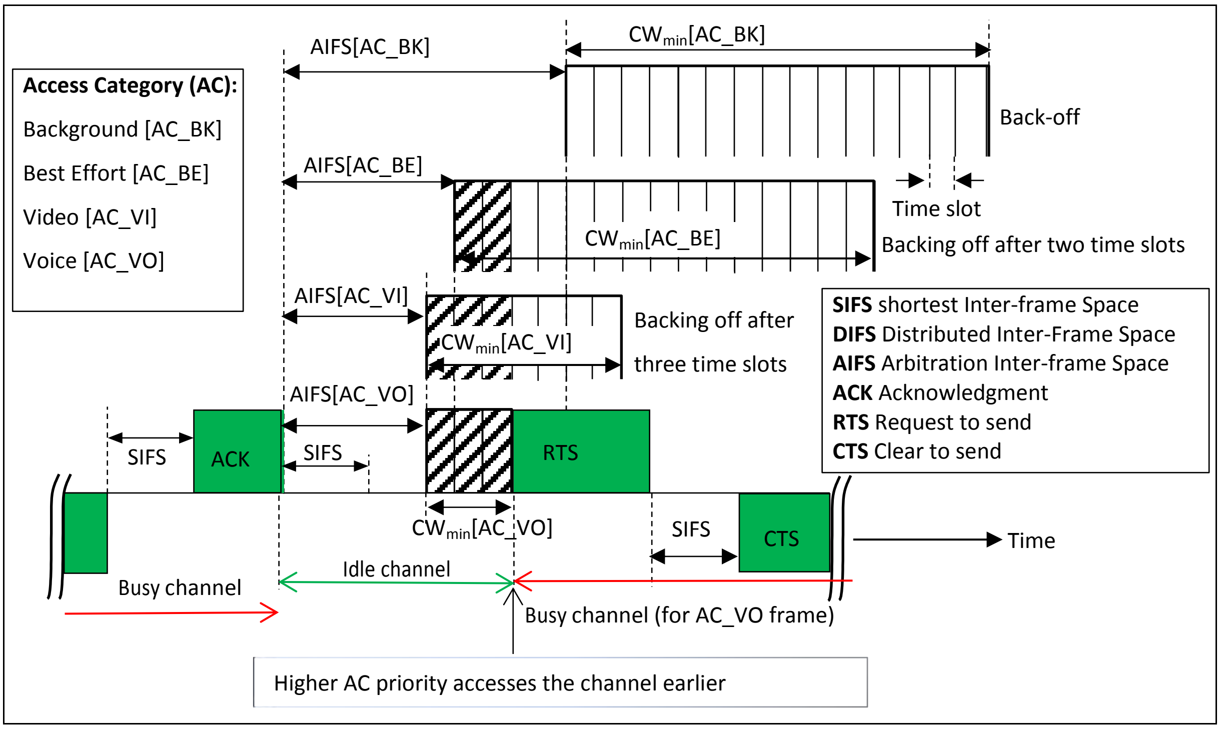


Figure 4.2 Channel access in IEEE 802.11e (EDCA) MAC

The TXOP is a new concept, introduced in IEEE 802.11e to limit the time of transmission for a given station. If a frame to be transmitted requires more than the TXOP interval, it should be fragmented into smaller frames that can each be transmitted within a TXOP interval. However, the use of a TXOP interval limits the number of smaller frames that may be aggregated [202]. Observing the IEEE802.11e standard is mandatory in new IEEE802.11 devices, including White-Fi devices. The IEEE 802.11e standard can handle coexistence with legacy stations, but with poorer QoS [203]. In the case of White-Fi, the sensing duration will impact on the effectiveness of IEEE802.11e in improving QoS in IEEE802.11 networks. The impact of the sensing operation should be investigated under different settings of the associated parameters, in particular, CW, AIFS, and TXOP. For White-Fi networks based on sensing, increasing the sensing duration for a more accurate outcome can compromise the IEEE 802.11e QoS mechanism, as shown by the simulation results in Section 5.3.

4.4. A QoS awareness sensing strategy solution for White-Fi

An SU may run different applications that have diverse QoS requirements. For instance, real-time applications are more sensitive to delay and jitter than email applications, and therefore, QoS requirements will depend on the applications running on a CR device and may vary during its operation. For better QoS, the sensing parameters should be adjusted in real time. The sensing operation has an impact on delay, achieved throughput, and packet loss, but the parameters of the sensing operation that affect QoS most are sensing duration, sensing frequency and sensing accuracy. These can be adjusted to mitigate degradation in QoS caused by any constraints in protecting the PU and achieving high spectrum utilisation. ED is commonly used for sensing on CSMA/CA-based networks because of its simplicity and low overhead, but in a CR environment its main drawback is the inability to distinguish PU signals from other signals, which reduces its effectiveness. The basic mechanism of CSMA/CA with and without RTS/CTS for CR is illustrated in Figure 4.3. The most challenging operations and decisions are highlighted (blocks A, B, C and D); all are extremely dependant on the accuracy of the sensing operation outcomes.

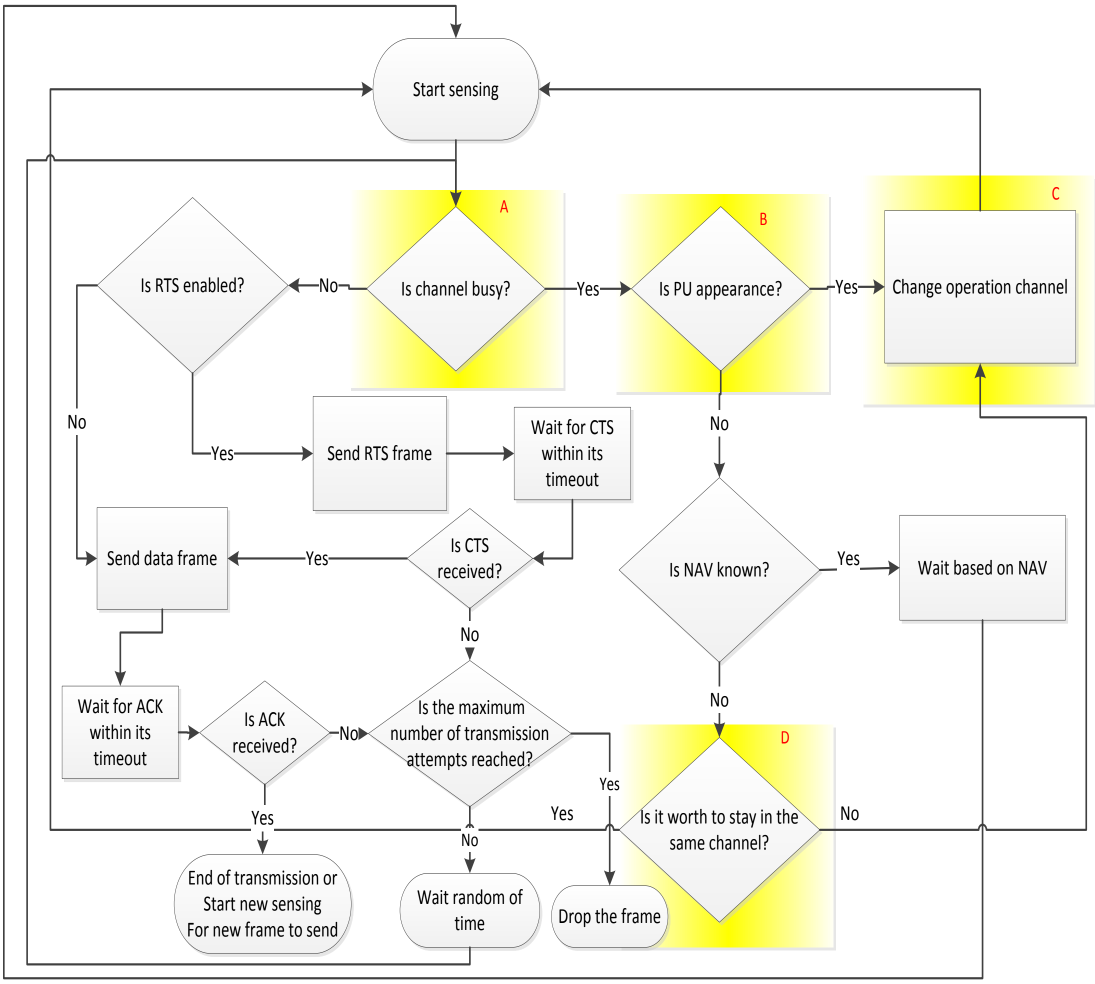


Figure 4.3 Basic flow chart of MAC protocol for White-Fi based on sensing

When using ED, any identification of the busyness of a channel can be quite accurate in high single-to-noise ratio (SNR) environments, but ED cannot provide adequate information for deciding the PU appearance (see Section 2.3). As the matched filter sensing technique can overcome this drawback and provide higher detection accuracy at the cost of more sensing time, complexity and power consumption, the sensing method should be selected during operation by trading between the sensing requirements and their implications. Such an approach requires that the White-Fi device support a set of different effective sensing techniques, each one suitable for a particular set of operational requirements. Other general factors that should be considered for selecting the proper sensing method have been discussed in Section 3.3. The main factors are the application QoS, CR device capabilities, the necessary protection for PU, and the available information about its signal.

In White-Fi, more specific factors should be considered based on the IEEE 802.11e requirements and possible operational settings, such as fragmentation, aggregation and RTS thresholds. The selected sensing method's parameters, such as sensing duration and interval, should be adjusted to achieve the best sensing accuracy with the minimum overhead. As a consequence, the sensing duration, $S_d[AC]$, should be varied based on the AC, without compromising PU protection, to maintain the prioritisation mechanism of the 802.11e standard. When AC_VI and AC_VO frames are to be sent, the sensing duration should be minimised as much as possible, or the AIFS[*AC*]variance becomes ineffective for long sensing times.

According to the IEEE standards for sensing in TV white space, the time required for both sensing and handoff operations should not exceed two seconds and the P_f and P_d should be less than 0.1 and 0.9 respectively [99]. Different sensing methods can achieve the required values for P_f and P_d under different SNR levels of the channel with various sensing durations. For example, ED method requires around 15 ms sensing duration at -20 dB SNR while the cyclostationary detection method needs around 50 ms at the same SNR [121, 204]. The selection of the proper sensing method should be performed in real time to meet the various requirements within these restrictions optimally, as proposed in the previous chapter. In this section, a sensing strategy is proposed for White-Fi to utilise a customised model of the general sensing selection mechanism proposed in Section 3.5.

4.4.1. Customise the proposed fuzzy logic selection mechanism for White-Fi

To enhance QoS, the proposed fuzzy inference system (FIS) mechanism of selecting the proper sensing method explained in the previous chapter is customised for White-Fi networks. The various applications' QoS requirements are the most important factor in selecting the sensing method. If each set of the specific application requirements is considered separately, this creates a large set of inputs, and lack flexibility. To avoid this, the required QoS for different applications is classified into four levels (see Table 3.1) and mapped to the four ACs in IEEE 802.11e with predefined input variables for the FIS, as shown in Table 4.2.

Table 4.2 Mapping the proposed application classes to EDCA frame access category

Application Class (Examples)	QoS level	Input variable	Frame Access Categories
Application A **Voice/Video conversation**	Very High	**1**	**Voice (AC_VO)**
Application B **Video streaming**	High	**0.666**	**Video (AC_VI)**
Application C **Email and Internet browsing**	Med	**0.5**	**Best effort (AC_BE)**
Application D **Internet relay chat**	Low	**0**	**Background (AC_BK)**

The IEEE 1900.6a™-2014 standard specifies four possible sensing methods that the CR device can use for sensing: ED, matched filter sensing, cyclostationary detection, and wideband sensing [205]. In this study, four sensing modes are defined to cover any other sensing techniques that can be used. The core attributes considered in this category are accuracy, the required sensing duration, and the ability to distinguish between observed signals. In each of these modes, any sensing technique that fits its characteristics can be used. The proposed four modes are Coarse, Moderate, Fine and Extra Fine. In this case, the White-Fi device is assumed to be capable of using different sensing techniques, including sophisticated techniques such as matched filter sensing.

The sensing method classification suggested in Table 3.2 is matched to the sensing modes proposed for White-Fi, as demonstrated in Table 4.3, as the terms used are easier for humans to understand. The input and output membership functions of the proposed FIS can be adjusted according to operational requirements. For simplicity, the settings of the FIS system follow the adjustments recommended in Section 3.4: Mamdani-type for inference, minimum for AND method and implication, maximum for aggregation and centroid for defuzzification.

Table 4.3 Matching sesning methods with the proposed sesning modes

Method class	Example	Matching to the proposed sensing modes in White-Fi
Method 1	Matched filter	**Extra Fine**
Method 2	Correlation or cyclostationary detection	**Fine**
Method 3	Covariance	**Moderate**
Method 4	Energy detection	**Coarse**

The input and output membership functions suggested in Figures 3.10–3.14 (see pages 72–76) are used. Prior information about the PU signal is assumed, and the necessary protection is low because PU signals in TV bands are typically non-sensitive applications such as TV broadcasts and wireless microphone. The general proposed FIS selection scheme still can be used if one, or more, of the assumptions used for customisation in this section is not valid. The values for the input variable application QoS set can be 0, 0.333, 0.666 or 1, resulting in an input subset membership of Low, Med, High or Very High, respectively and explicitly. Accordingly, each value is given to one of the four possible AC frames, as shown in Table 4.2. The input variable for prior information is fixed at 1 (so that the input set of this variable is a member only of High) and 0 (so the input set of this variable is a member only of Low) for the required PU protection input variable. For CR capability, the input variable is set at 1 (so that the input set of this variable is a member only of High). The rules that satisfy this special case of assumptions are highlighted in Table 4.4 (only the related rules are shown from all the rules listed in Table 3.3). All rules are set to the same regular weight, which is the value one. The implementation snapshots of the proposed FIS using MATLAB and the resulting selection method based on the given input variables are available in Appendix A. Under the above -mentioned settings, the proposed FIS will select different sensing methods—that is, modes—based on the frame AC, which is equivalent to the simple selection illustrated in Figure 4.4

Table 4.4 Applied FIS rules for the customised White-Fi scenario

Rule	**If Input (connection between input variables is AND)**				**Then Output**
	Application QoS (value)*	***CR capability (value)****	***Prior information (value)****	***The required PU protection (value)****	***Method class (value)****
2	Very High (1)	High (1)	High (1)	Low (0)	Method 4 (0.785)
20	High (0.666)	High (1)	High (1)	Low (0)	Method 3 (0.623)
37,38	Med (0.333)	High (1)	High (1)	High, Low (0)	Method 2 (0.375)
55, 56-58	Low (0)	High (1)	High (1), Med	High, Low (0)	Method 1 (0.125)

**The specific chosen values for the input variables and the resulted output are highlighted*

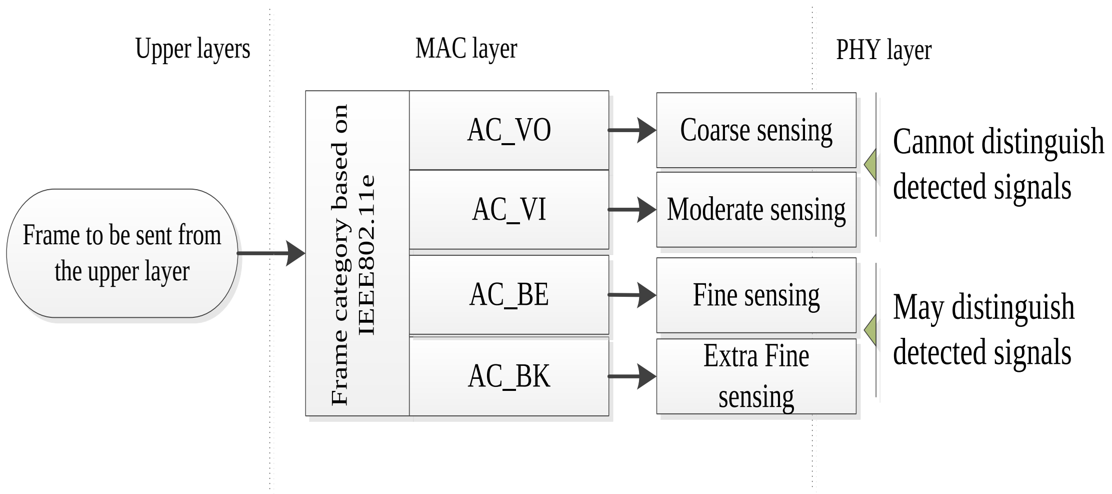


Figure 4.4 Sensing selection mechanism based on frame category

Integrating the proposed selection mechanism into the MAC protocol of White-Fi is another challenge that differs according to the assumptions used and to what extent they are reasonable. A QoS Awareness CR MAC protocol (QACR-MAC) for White-Fi is proposed in which the sensing selection mechanism is used under the aforementioned settings and assumptions. Basically, before the first transmission attempt, each sensing mode will be used based on the AC of the frame to be transmitted, as shown in Figure 4.4.

4.4.2. QACR-MAC

In this section, the proposed QACR-MAC protocol is described in detail. The protocol is based on integrating the sensing selection mechanism into the EDCA, which is a mandatory feature in the IEEE 802.11af standard. In this proposed solution, the spectrum assessment does not rely on GDB, rather, it relies on the available sensing techniques with consideration of their capabilities and limitations. Other design goals and criteria are listed below:

- No GDB is required.
- No CCC is required.
- No coordination is required for MAC (contention access based) suitable for ad-hoc networks.
- No underlay access is assumed.
- No cooperation or synchronisation between PU and SU systems are required.

- Compliance with the current IEEE standards for CR, e.g., 802.11af is required.
- Maintaining the efficiency of IEEE 802.11e standard and enhancing the overall QoS is required.
- The use of high-accuracy spectrum sensing when there is adequate time to conduct it, for greater PU protection and frequency spectrum utilisation, is desirable.
- Using low sensing durations when accuracy may be compromised, for higher priority access, is acceptable.
- Reducing the required scan and handoff between spectrum white spaces is preferable.

To achieve the desired design goals, several reasonable assumptions are considered in the proposed QACR-MAC. As mentioned in Section 4.4.1, the main three assumptions are listed below:

1. That the SU device can use at least one sensing technique that satisfies the requirements for each proposed sensing mode, Coarse, moderate, Fine and EF. Thus, at least four different sensing techniques are supported.
2. That necessary prior information about PU signals for the supported sensing techniques is available.
3. That the PU systems are not being used for sensitive applications that require high protection.

The proposed sensing duration ranges are divided by three thresholds: $\mathbf{Thr_1}$,$\mathbf{Thr_2}$ and $\mathbf{Thr_3}$, as shown in Table 4.5 and defined below:

$$S_d[AC\_VO] \le \mathbf{Thr_1} < S_d[AC\_VI] \le \mathbf{Thr_2} < S_d[AC\_BE] \le \mathbf{Thr_3} < S_d[AC\_BK] \le \mathbf{S_{dMAX}} \qquad (4.2)$$

Table 4.5 Sensing strategy based on frame access category

Frame Access Categories	Sensing mode	Sensing duration (S_p)	Sensing characteristics
Voice (AC_VO)	Coarse sensing	$S_d[AC\_VO] \leq \mathbf{Thr_1}$ E.g., $\mathbf{Thr_1}$ =1ms	Only blind sensing can be used. Cannot distinguish between PU and SU; hence, no handoff. Poor performance in low SNR.
Video (AC_VI)	Moderate sensing	$\mathbf{Thr_1} < S_d[AC\_VI] \leq \mathbf{Thr_2}$ E.g., $\mathbf{Thr_2}$ = 5ms	More advanced sensing methods but still not capable of distinguishing between PU and SU. Moderate sensing accuracy even in low SNR.
Best effort (AC_BE)	Fine sensing	$\mathbf{Thr_2} < S_d[AC\_BE] \leq \mathbf{Thr_3}$ E.g., $\mathbf{Thr_3}$ =50ms	Some sensing methods that can distinguish between PU and SU can be used; hence, handoff is conducted when PU appearance is recognised. High sensing accuracy.
Background (AC_BK)	Extra Fine sensing	$\mathbf{Thr_3} < S_d[AC\_BK] \leq \mathbf{S_{dMAX}}$	Sensing methods that can distinguish between PU and SU can be used; hence, handoff is conducted when PU appearance is recognised. High sensing accuracy even in low SNR.

The sensing duration should not be less than the minimum time required for sensing one channel, $\mathbf{S_{dMIN}}$, and not larger than the maximum allowed duration $\mathbf{S_{dMAX}}$. Theoretically, $\mathbf{S_{dMIN}}$ is the time slot, aSlotTime, defined in IEEE 802.11 2012 based on the modulation at the physical layer.

$$\text{aSlotTime} = \text{aCCATime} + \text{aRxTxTurnaroundTime} + \text{aAirPropagationTime} + \text{aMACProcessingDelay} \quad (4.3)$$

where aCCATime is the time required to conduct CCA-ED. The aRxTxTurnaroundTime is the time for the transceiver switch between transmitting and receiving states. The propagation delay is represented by aAirPropagationTime and the time required for processing CCA-ED at the MAC layer is aMACProcessingDelay.

$\mathbf{S_{dMAX}}$ is calculated based on the time required for scan and handoff, T_{HO}, to another available white space channel within the regulation limit of 2 Seconds as below:

$$\mathbf{S_{dMIN}} \leq S_d[AC] \leq \mathbf{S_{dMAX}} = T_{HO} - 2 \text{ sec} \tag{4.4}$$

$\mathbf{Thr_1}$ could be equal to the required time slots for conducting $\mathbf{S_{dMIN}}$, i.e., $\mathbf{S_{dMIN}}$ (aSlotTime) or its multiplications, as follows:

$$\mathbf{S_{dMIN}} \text{ (aSlotTime)} \leq \mathbf{Thr_1} \text{ where } \mathbf{Thr_1} = R_1 \times \text{aSlotTime} \tag{4.5}$$

Thus, the other thresholds are defined as follows:

$$\mathbf{S_{dMIN}} \leq \mathbf{Thr_1} < (R_2 \times \text{aSlotTime} = \mathbf{Thr_2}) < (R_3 \times \text{aSlotTime} = \mathbf{Thr_3}) \leq \mathbf{S_{dMAX}} \tag{4.6}$$

where R_1, R_2 and R_3 are integers and $R_1 < R_2 < R_3$. (4.7)

These integers are selected, so each range provides adequate time for its sensing methods. For more flexibility, the values of these integers are adjusted according to the design requirements. The minimum value of each integer is the number of time slots required for the sensing mode of that range to sense the current operating channel/channels. The maximum values for R_1, R_2 and R_3 are constrained by the maximum values that can be chosen to satisfy equations 4.5 and 4.6. While the minimum values, especially for R1 and R2, help to reduce the delay caused by sensing, increasing them is required either for better accuracy or to sense more channels, leading to greater spectrum utilisation. The number of operational channels, N_{op}, and the number of channels to sense, N_s, are essential factors for calculating R_2, R_2 and R2. In TV white space, N_{op} could be 1, 2 or 4 (contiguous or non-contiguous). The maximum number of channels to be sensed, N_s, is limited to TV bands based on different regulations, so the minimum and maximum values of R_1, R_2 and R_3 are defined as follows:

R_{1MIN} = N_{op} × number of **time slots required by Coarse sensing for one channel** (4.8)

R_{1MAX} = maximum available N_s × number of **time slots required by C sensing for one channel** (4.9)

R_{2MIN} = N_{op} × number of **time slots required by Moderate sensing for one channel** (4.10)

R_{2MAX} = maximum available N_s × number of **time slots required by Moderate sensing for one channel** (4.11)

R_{3MIN} = N_{op} × number of **time slots required by Fine sensing for one channel** (4.12)

R_{3MAX} = maximum available N_s × number of **time slots required by Fine sensing for one channel** (4.13)

In the Coarse sensing mode, the sensing duration $S_d[AC]$ is less than or equal to **Thr₁** to give higher priority to AC_VO frames with less impact on the delay. However, as the sensing methods that can be used within this short time are blind, such as ED, the Coarse sensing mode cannot distinguish between PU and SU signals, and poor accuracy is expected at low SNR. Under the Moderate sensing type, the sensing methods that can be used for an $S_d[AC]$ larger than $\mathbf{Thr_1}$ up to $\mathbf{Thr_2}$ are similar to those used in the Coarse sensing mode. However, they may have a slight improvement in sensing accuracy particularly at low SNR. The AC_VI frames have less priority than AC_VO to win the contention window, and thus more delay is predicted. For the Fine sensing mode, the sensing duration $S_d[AC]$ is larger than $\mathbf{Thr_2}$ up to $\mathbf{Thr_3}$, so sensing methods that can differentiate PU from other signals can be used as more time is available for sensing. The Fine sensing mode enables more utilisation of white spaces, but causes higher delay, as shown by the simulation results in the next chapter. Furthermore, conducting Fine sensing before AC_BE frames maintains the desired priority of these frames. As AC_BK frames have the lowest priority, Extra Fine sensing should be carried out before them for a sensing duration $S_d[AC]$ larger than $\mathbf{Thr_2}$ but not exceeding $\mathbf{S_{dMAX}}$.

In the Extra Fine sensing mode, sophisticated sensing methods can be used to achieve high sensing accuracy even under low SNR, so that higher spectrum utilisation can be achieved with a cost of higher delay. When Fine and Extra Fine sensing recognise the appearance of a PU in the current white space channel, the White-Fi device must scan for other vacant channels and leave the currently occupied one. When the current channel is found to be occupied only by other SUs, the device can continue to use and share the channel. For efficient sharing of available spectrum holes among different SU systems, widely adopted coexistence rules and protocols should be imposed. The $S_d[AC]$ is conducted before sending the frame for the first attempt as a substitute to the AIFS[*AC*]. Then it follows the conventional behaviour of EDCA for extra attempts, as illustrated in Figure 4.5.

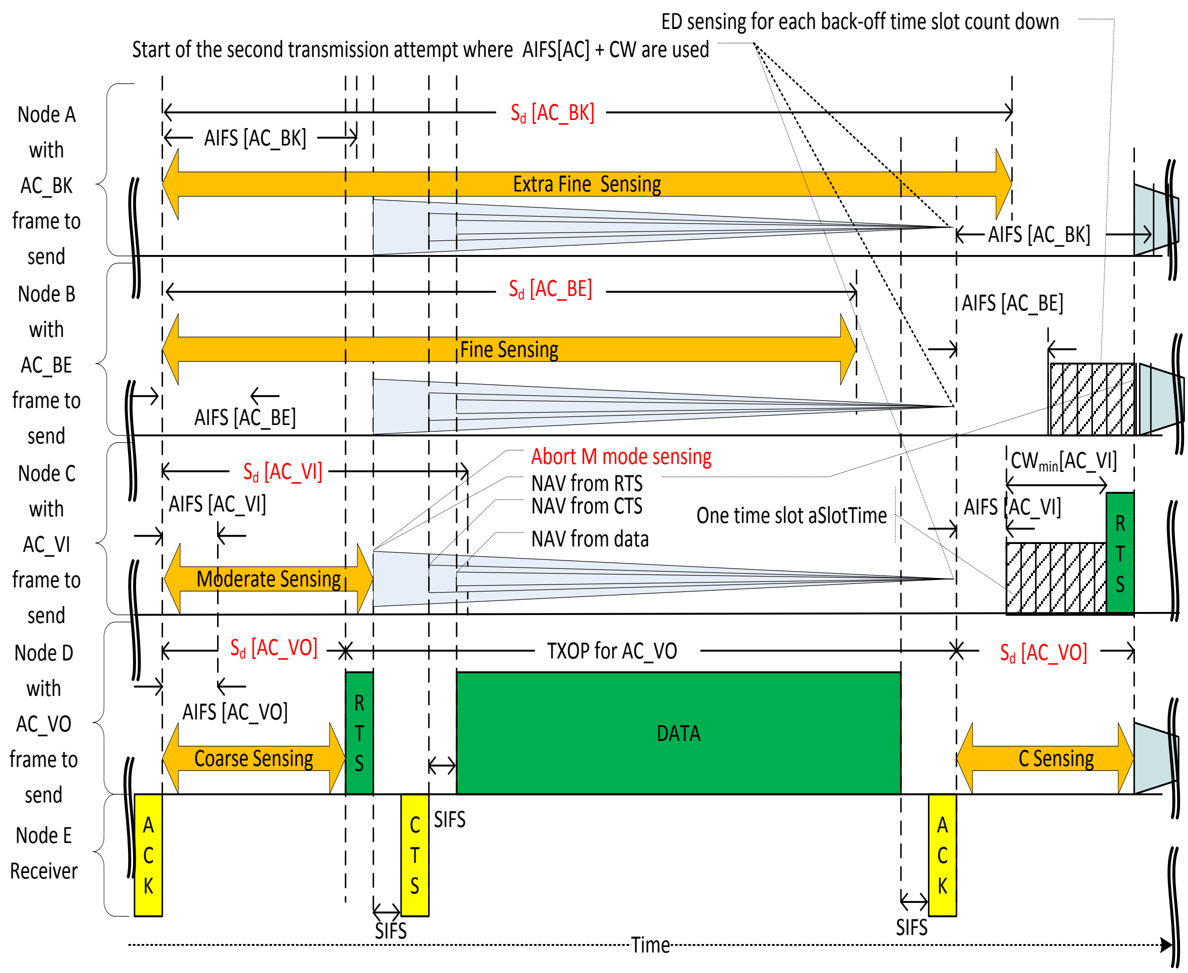


Figure 4.5 Sensing strategy in QACR-MAC when four nodes contend to send different AC frames (first transmission attempt)

During S_d[*AC*] the device can listen to any transmitted frames but cannot transmit any in the same channel. When the NAV is identified during S_d[*AC*], the node acts based on its frame AC. For Coarse and Moderate modes, during S_d[AC_VO] or S_d[AC_VI], after AIFS[*AC*] elapses the node aborts the sensing once NAV is received and starts deferred access mode until the last updated NAV countdown reaches zero (see Node C in Figure 4.5). For Fine and Extra Fine modes, the S_d[AC_BE] and S_d[AC_BK] are not aborted even if NAV is identified (see nodes A and B in Figure 4.5). If no NAV is identified while sensing, the node sends an RTS/data frame once its sensing duration has elapsed and the channel is identified as idle. However, if the NAV is identified during S_d[AC_BE] or S_d[AC_BK], the node starts a countdown NAV count and then AIFS[*AC*] and back-off procedures, based on EDCA. This may cover one contention window or more during S_d[AC_BE] or S_d[AC_BK]. Hence, when the channel identified as idle at the end of sensing Fine or Extra Fine, the node will not send RTS/data until the end of the current contention window and its back-off timer.

The node after the first attempt at sending a frame will use the conventional CCA, where ED will be used for each time slot interval until sending or dropping the frame. For example, in Figure 4.5, node A has to wait for the remaining AIFS[AC_BE]+back-off after the end of its S_d[AC_BE] and node for the remaining NAV+ AIFS[AC_BE]+back-off. Node D has to sense for S_d[AC_VI] (as this is its first attempt to send this new AC_VI frame). In tnis case, node C wins the channel as its AIFS[AC_VO]+back-off (second attempt for sending AC_VO frame) elapsed first. The back-off procedure is used to reduce collision between contended nodes. After the first attempt the node chooses a random back-off time slot within the CW range, from 0 to CW_{min}[*AC*] as defined in Table 4.1. For instance, node C in Figure 4.5 chose the random back-off to be the minimum value of CW: typically CW_{min}[AC_VI] =7 time slots. During the back-off stage, if a node finds the channel is busy, this result is decided by ED based on CCA-ED, the node freezes its back-off countdown and adds this attempt to the unsuccessful retransmission counter for that frame. When the channel is identified as idle, the node resumes the last back-off counter and repeats this procedure until the back-off reaches zero, when it will send the frame. If the retransmission attempts reach the maximum allowed number the frame is dropped from the transmission queue.

4.4.2.1. QACR-MAC when the PU is active

When sensing Coarse and Moderate, the presence of PU in the current channel cannot be known for certain, and as long this is so other factors should be considered before conducting scans and handoff procedures. The scan and handoff cover the spectrum of mobility, decision and sharing CR functions. The busyness of the channel, in such blind sensing, could be caused by the transmission of a PU signal, other White-Fi signals (collision), other SU signals, or noise (false alarm). The collision with other White-Fi signals may be recognised in a relatively short time unless there are one or more hidden nodes. The RTS/CTS could help by reducing this problem, but it is not yet a mandatory feature in 802.11af and the 802.11 family. It is optional, and can be enabled based on the transmission frame length. The maximum transmission time regulation, T_{MaxTx}, is an important limit for SU systems. In the case of White-Fi, the TXOP limits the time allowed for each AC to transmit frames when it has won a contention window. The maximum duration for transmitting one frame, aPPDUMaxTime, is 20 ms [65]. The TXOP limit in 802.11af is 22.56 ms (for BCU 6 and 7 MHz; see Table 4.1), so it can be considered the T_{MaxTx} for White-Fi. Hence, the maximum possible collision period is 22.56 ms because of hidden nodes. For sensing one channel, $S_d[AC]$ is less than 22.56 ms for C. However, if the $S_d[AC] \geq T_{MaxTx}$ and the channel are found busy for C and Moderate modes, SU should assume that a PU signal exists and start the scan and handoff procedure. For the C and Moderate modes, when $S_d[AC] < T_{MaxTx}$ and no NAV can be identified and the channel is still busy, the node can use the last known NAV to defer its access until a new NAV is received from contending neighbours. If no new NAV is received during the used NAV, the node keeps sensing the channel based on CCA-ED, seeking idle states. When the total time of a busy state exceeds a maximum waiting time $T_{MaxWait}$, the node should start the scan and handoff procedure, as shown for nodes C and D in Figure 4.6. The $T_{MaxWait}$ minimum time is proposed to be SIFS + T_{MaxTx}. Based on the FCC regulations, the $T_{MaxWait}$ must satisfy this relation:

$$T_{MaxWait} \leq (T_{HO} - 2 \text{ sec}) \qquad (4.14)$$

The $T_{MaxWait}$ should be larger than $S_d[AC\_BK]$ + time required to receive RTS, so there is a possibility of receiving RTS from a neighbour node that is sensing with higher accuracy. However, to comply with the regulations, $T_{MaxWait}$ cannot be larger than $\mathbf{S_{dMAX}}$ (see equations

4.4 and 4.14). In the case of Fine and Extra Fine modes, the sensing techniques used are able to distinguish PU signals from other SU signals with an accuracy of detection, P_d, mainly based on the operational channel conditions and length of $S_d[AC]$. A node can rely on the outcome of Fine and Extra Fine sensing and conduct the scan and handoff procedure if a PU transmission is identified, as shown for nodes A and B in Figure 4.6. If the channel is identified as idle and no NAV is received during an S_d[AC_BE] or S_d[AC_BK], the node can immediately send an RTS frame, as shown in Figure 4.7 (see node B). When a NAV is received during an S_d[AC_BE] or S_d[AC_BK], the node defers its access accordingly (see nodes A and B in Figure 4.7). As mentioned before, after the first attempt if the channel is found busy with other SUs and no PU signal is identified, the node uses the conventional EDCA mechanism until the frame is sent or dropped. In the case of Fine and Extra Fine sensing, if they cannot identify a PU signal but the channel is found busy and no NAV is received, then the same rules as for Coarse and Moderate modes are used, and the node starts the scan handoff procedure if the $S_d[AC] \geq T_{MaxTx}$ or waits for $T_{MaxWait}$ to be exceeded if $S_d[AC] < T_{MaxTx}$.

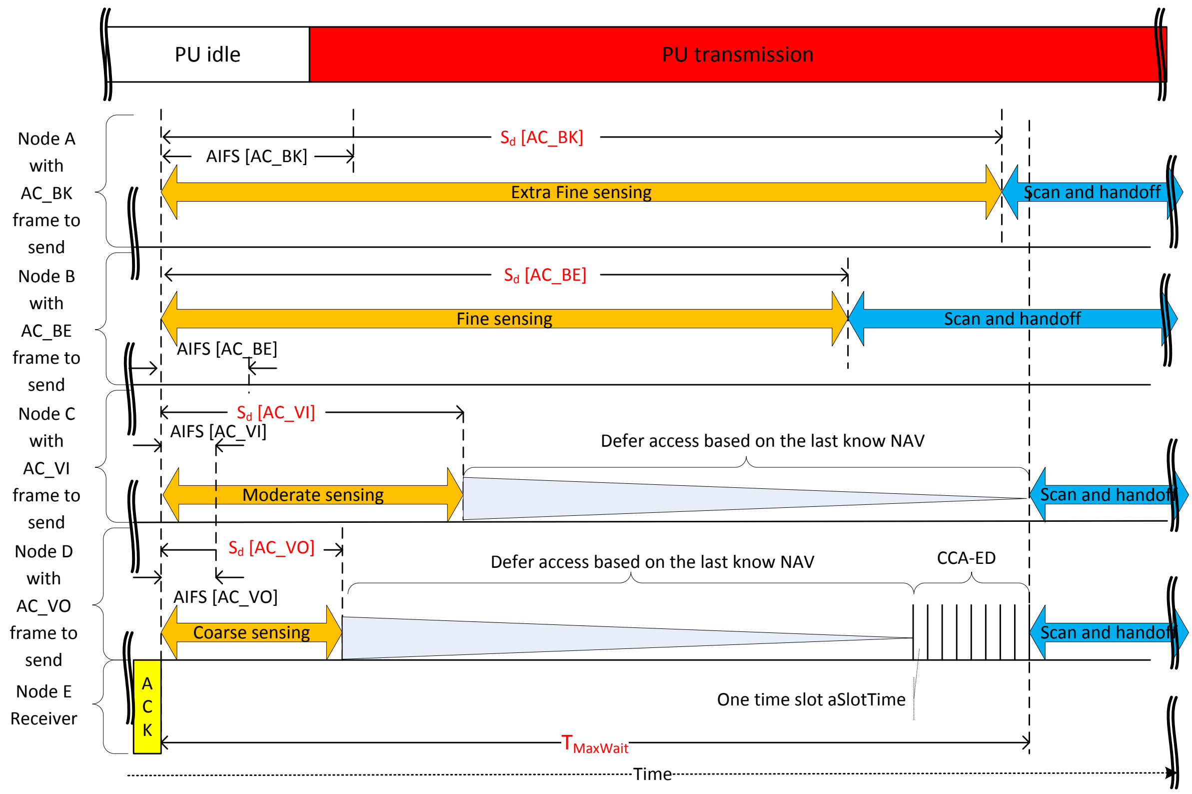


Figure 4.6 QACR-MAC mechanism during long PU transmission (first transmission attempt)

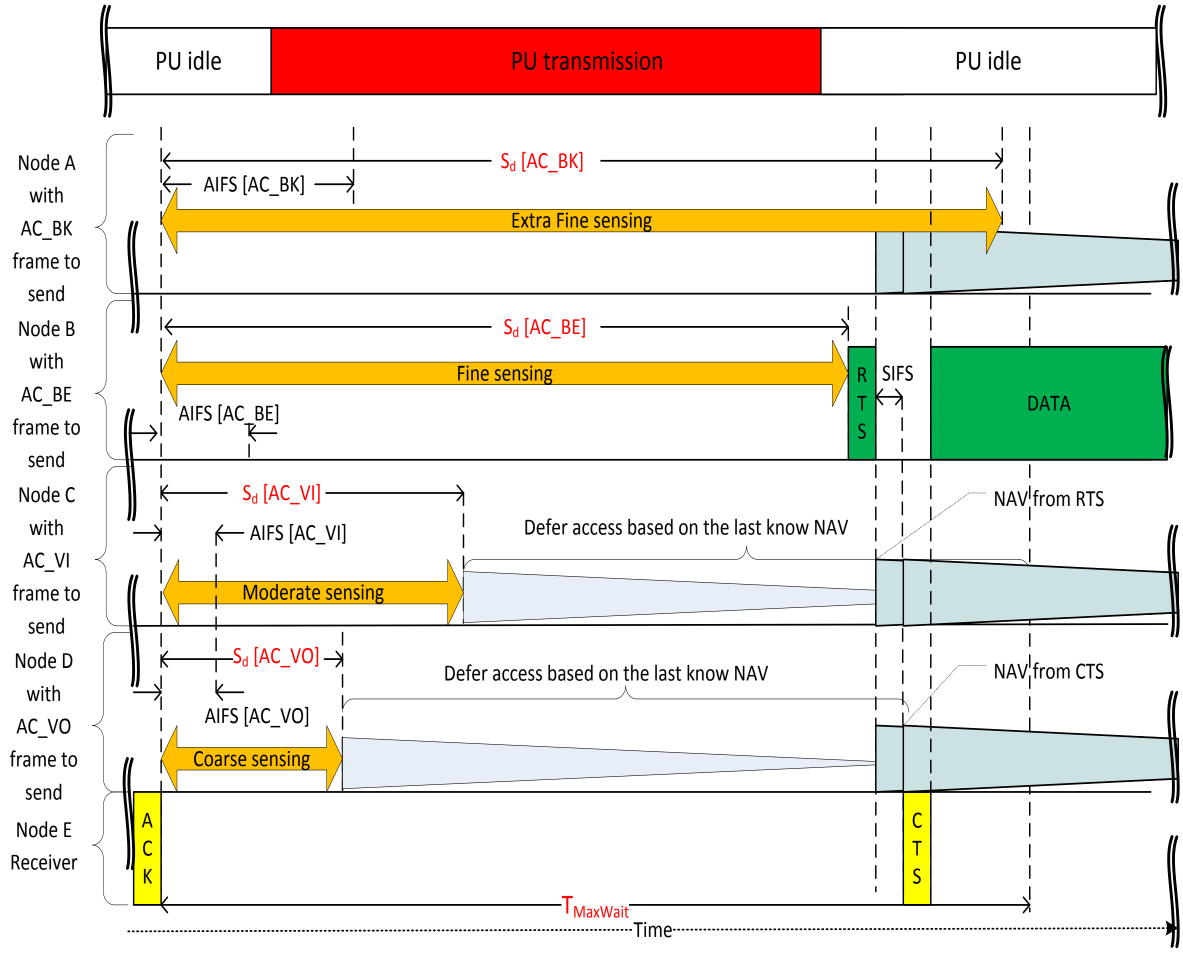


Figure 4.7 QACR-MAC mechanism for short PU transmission (first transmission attempt)

4.4.2.2. QACR-MAC after an external collision

An external collision occurs between nodes sharing a transmission channel. In contrast, an internal collision occurs between different AC frames of the same node contending in the transmission queue to access the channel. As an internal collision is not directly affected by PU activities, it is handled by the conventional EDCA in the proposed QACR-MAC. In this work, when collision is mentioned without specifying internal or external, it refers to an external collision. Generally, 802.11 wireless nodes, including White-Fi nodes, cannot recognise collisions during transmission. The RTS/CTS and ACK mechanisms are used at the MAC layer to detect the consequences of such collisions. In EDCA, the retransmission and binary exponential pack-off procedures are used to deal with the collision issue. If the transmitter node does not receive CTS for its RTS or ACK/B-ACK for data frames after a predefined timeout for each response, a collision is assumed. Then the node doubles its last CW and chooses a new random back-off in the new CW, and the retransmit procedure is conducted when the new back-off has elapsed. This binary exponential back-off procedure continues until the doubled CW reaches its maximum values, $CW_{max}[AC]$, with regard to the AC. Afterwards the node keeps using the $CW_{max}[AC]$ until it sends or drops the frame; that is, when the maximum allowed retransmission attempts is reached.

In a CR environment, the PU may cause a collision problem when it starts transmission during an ongoing frame exchange between other White-Fi nodes. The TXOP is proposed to limit the permitted duration for transmitting frames based on the AC of the frames. The source node in a TXOP can tell that there is a collision problem in the operating channel only when no response frame, e.g., CTS or ACK/B-ACK, is received. Though, it will not know if the problem is because of PU interference, collisions with other SUs, or wireless channel issues such as fading and attenuation. Such an instance should trigger more accurate sensing, rather than CCA-ED, to identify the situation. In QACR-MAC, the destination node should use the best possible sensing mode that the new length of back-off time allows, regardless of the AC of the current transmission attempt. Otherwise, CCA-ED is used. After a collision is detected, the new selected back-off number, $N_{back\text{-}off}$, of time slots is measured to see if it offers any better sensing mode and then sensing is conducted within the back-off time. Meanwhile the procedure of back-off timing is normally conducted, as per the IEEE 802.11e standard. Other sensing modes may be used instead of CCA-ED, for the given back-off time slots, if the duration is sufficient. If there are any remaining time slots in the back-off after conducting one of the other sensing modes, then CCA-ED is used for the remaining slots. However, when a NAV value is received during this stage, the node freezes the back-off counter and defers its access according to the received NAV value. The general equation for calculating the waiting timeout for response frames is:

$$\text{Response Timeout} = \text{SIFS} + \text{Response Transmission Duration} + \text{aSlotTime} \quad (4.15)$$

'Response Transmission Duration' is determined based on the response frame length, propagation delay and the physical rate expected based on the rules in IEEE 802.11 standard. The waiting timeout for ACK and CTS frames is the same as ACK length or CTS length. The collision back-off algorithm used when the response timeout is exceeded without a response is summarised as follows:

Algorithm 4.1

1: When waiting time for response > Response Timeout;

2: If (CW < CW_{max}[*AC*]) Then {double the size of the CW;}

3: Else {CW = CW_{max};}

4: Chose a new random $N_{back\text{-}off}$ within the range [0-CW];

5: If (S_d[AC_BK] ≤ $N_{back\text{-}off}$ × aSlotTime) Then {use S_d[AC_BK]; Go to 10;}

6: If (S_d[AC_BE] ≤ $N_{back\text{-}off}$ × aSlotTime) Then {use S_d[AC_BE]; Go to 10;}

7: If (S_d[AC_VI] ≤ $N_{back\text{-}off}$ × aSlotTime) Then {use S_d[AC_VI]; Go to 10;}

8: If (S_d[AC_VO] ≤ $N_{back\text{-}off}$ × aSlotTime) Then {use S_d[AC_VO]; }

9: Else {use CCA-ED; }

10: While ($N_{back\text{-}off}$ > 0)

11: If (channel is busy) Then freeze the back-off $N_{back\text{-}off}$ = $N_{back\text{-}off}$;

12: Else $N_{back\text{-}off}$ = $N_{back\text{-}off}$ -1;

13: End While;

14: Transmit the frame and wait for a response;

15: If (the frame successfully transmitted or dropped) Then CW = CW_{min}[*AC*];

16: Else Go to 1;

17: Exit

The highlighted steps 5 to 8 in algorithm 4.1 can imply the use of the proposed FIS selection mechanism by setting the suitable CR capability variable input for each available time, and also by considering other input sets if they have experienced any changes. However, for simplicity, the QACR-MAC is described based on this simple selection criterion that considers only available time.

Figure 4.8 illustrates an example of a collision between SUs when the PU is assumed idle and there are four nodes, A, B, C and D, contending in the same channel to send data to E. All nodes have RTS/CTS enabled and each node attempts to send a different AC frame. In this scenario, a collision occurs between RTS frames of A and D so receiver node, E, will not send back CTS. The collision will be recognised by nodes A and D after CTS timeout, and they will wait for AIFS[*AC*] and then perform the collision back-off algorithm. It is assumed that both A and D achieved their CW_{max}[*AC*] and have chosen $N_{back\text{-}off}$ to be equal to CW_{max}[*AC*]. For node A, the $N_{back\text{-}off}$ = CW_{max}[AC_BK] = 1024 × aSlotTime, which is about 24.576 ms. When Thr_3 = 50 ms, Extra Fine sensing cannot be conducted. As S_d[AC_BE] could be less than 24.576 ms, Fine sensing is conducted as shown for node A.

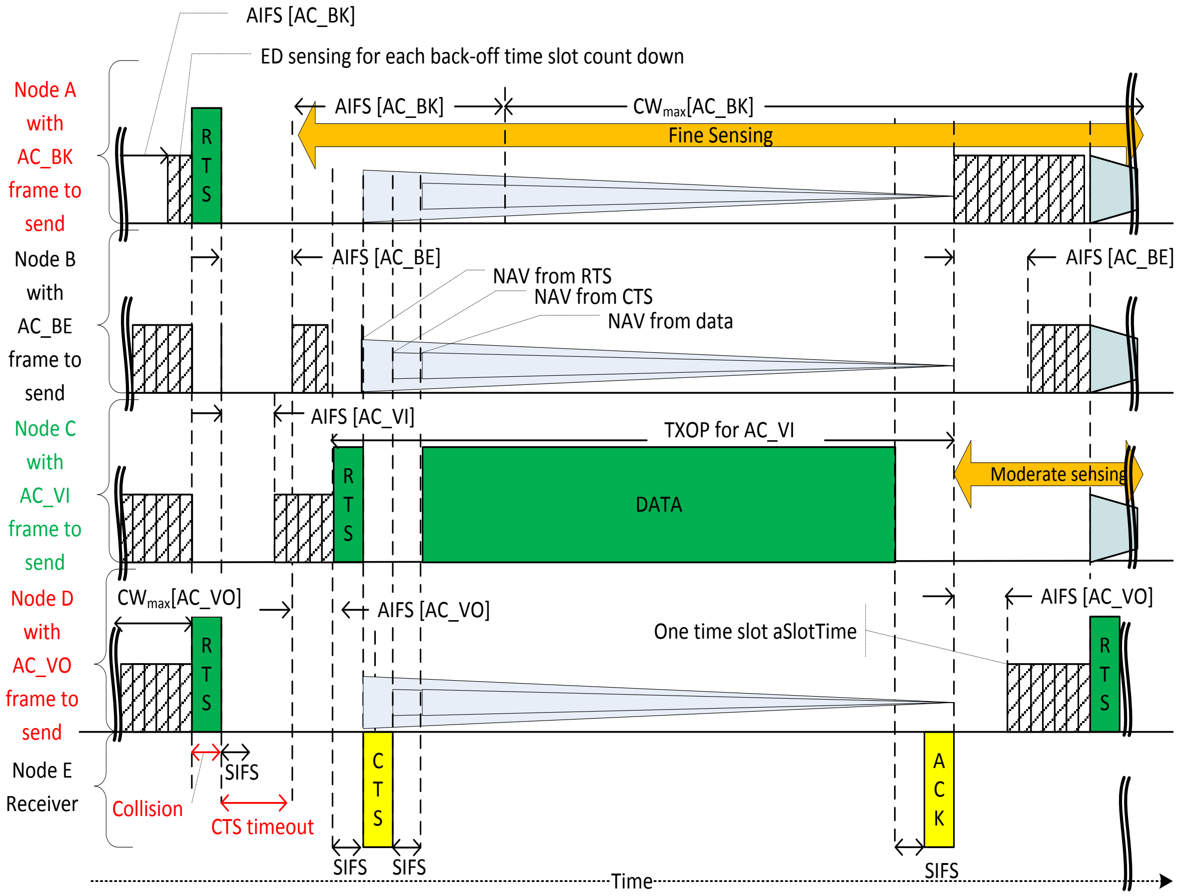


Figure 4.8 Collision between SUs when the PU is idle

For node D, the $N_{back-off}$ = CW_{max}[AC_VO] = 7 × aSlotTime, which is about 0.168 ms; hence, when Thr_3 = 1 ms, only Coarse sensing can be conducted. Simply, node D will use CCA-ED as normal and wait for AIFS[AC_VI] + $N_{back-off}$, as shown in Figure 4.8 for node D. After the collision, nodes B and C will wait for AIFS[*AC*] and then resume their conventional back-off procedure. Node C with AC_VI frame wins this contention as its remaining back-off is less than that of node B. An example of a collision caused by PU transmission is illustrated in Figure 4.9. In this example, a node with AC_BK frame to send, node A, wins the contention as its AIFS[AC] + remaining back-off is less than the other contending nodes, B, C and D. The RTS frame is sent successfully to node E and the other nodes, B, C and D, defer their access according to the received NAV. After an SIFS, node E responds with a CTS frame, but the PU starts transmitting during CTS transmission so CTS will not be received successfully within the CTS timeout. Node A applies the collision back-off algorithm as in the previous example and starts Fine sensing. After the NAV counter elapses, the other nodes use CCA-ED to assess the channel for each intervening time slot until the channel is identified as idle or $T_{MaxWait}$ is passed. In this scenario the PU transmission is short, less than $T_{MaxWait}$, and consequently no node triggers the scan and handoff procedure.

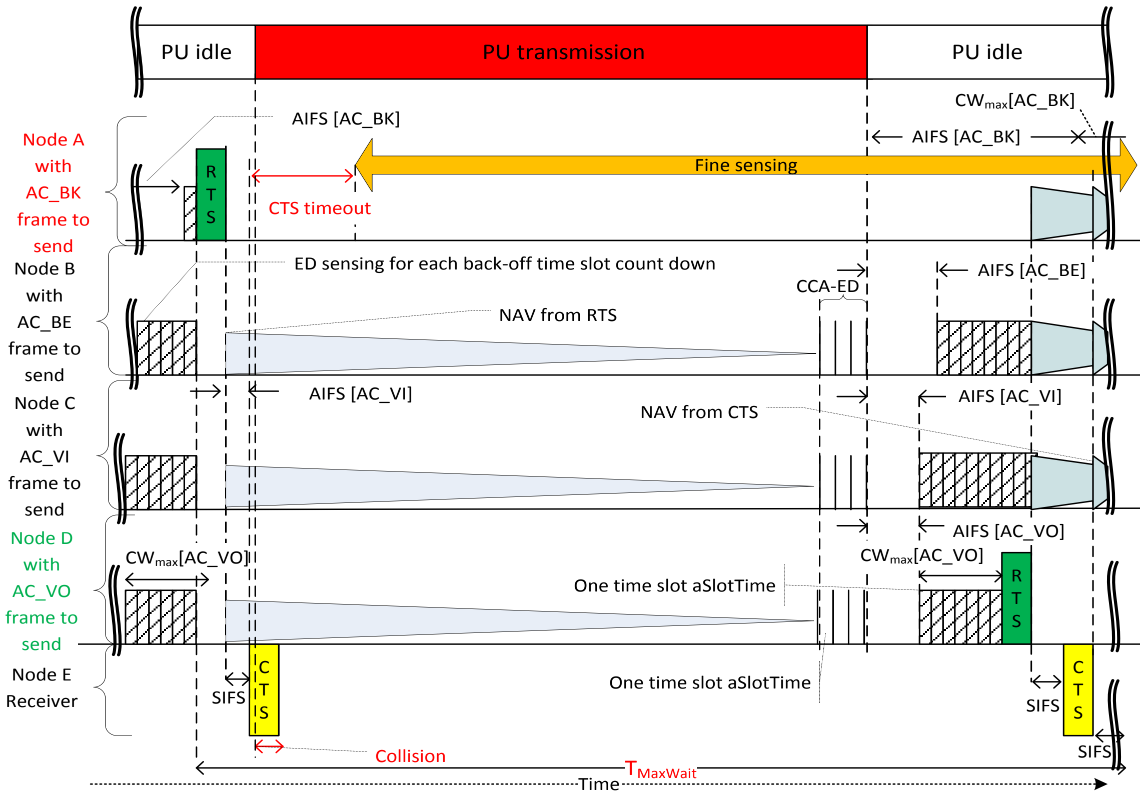


Figure 4.9 Collision with PU

4.4.2.3. QACR-MAC and scan and handoff

Spectrum handoff between the available white spaces is required in CR environments as discussed in 1.1.2. Handoff is also required in ISM bands when a mobile IEEE 802.11 node roaming between different APs is configured with different operational channels. Before the node completely hands off and changes its operation channel, a scan is conducted on available channels to find beacon or probe-response frames of its network. The process of scanning and handoff causes a noticeable delay that changes depending on the number of channels to be scanned and the scanning method used. In the Wi-Fi environment, a scan and handoff are rarely required unless there is an extremely mobile node that moves frequently between coverage of different APs. In the case of CR, a handoff is triggered more frequently proportional to PU activities. Spectrum mobility and decision are other areas in CR studies that address how efficiently SUs scan and handoff to a new operational channel within the available white spaces. In this study these areas are not addressed, and the assumption is that the conventional scanning and handoff procedures of IEEE 802.11 standards without GDB are used.

The conventional scan procedure includes active and passive scanning. Passive scanning is a mandatory feature whereby an AP periodically broadcasts a beacon frame in the operational channel, typically every 100 ms. Nodes belonging to the same network scan the available channels for their network beacons and in each channel waits up to the beacon's interval time, e.g., 100 ms. The maximum passive scanning delay is determined by the number of channels to be scanned and the beacon's interval:

$$\text{Maximum passive scanning delay} = \text{number of channels } N_s \text{ * beacon's interval} \qquad (4.16)$$

For instance, the FCC regulation provides 30 non-overlapping TV white space channels of eight MHz bandwidth each [61]. The maximum passive scanning delay is 3000 ms (3 seconds) when N_s is equal to 30, which is typically the maximum number of channels in TV white space, and the beacon interval is 100 ms (default).

In active scanning (optional), instead of waiting for beacons a node broadcasts a probe-request frame in the current channel and waits for probe-response frames. The node waits at least MinChannelTime in idle channels up to MaxChannelTime when activity is detected before it moves to another channel [206]. Typically the maximum time for a node to wait for response frames is 60 ms; if there is no response, it waits for a minimum 20 ms before it moves to another channel [207]. Hence, the total maximum scanning delay is reduced in the active method for the same N_s but at the cost of more bandwidth use by management frames: the probe request and probe response. In CR environments, PU activities and avoiding interference with them are new factors to be considered in scan and handoff issues.

The common goals in designing CR MAC protocols are to reduce scanning time and avoid unnecessary handoff attempts. When there is no up-front information about PU activities, sensing is the main tool for detecting PU activity and triggering handoff. The QACR-MAC is designed to reduce unnecessary scan and handoff procedures, and a channel will be vacated when PU is detected, or if the channel is busy for longer than $T_{MaxWait}$ and there is no indication that it is busy merely because of SU use. Only in Fine and Extra Fine sensing modes the scan and handoff procedures are triggered when a PU is detected, based on the high-accuracy spectrum assessment techniques they use. In contrast, no handoff is conducted if the channel is found busy by conducting Coarse or Moderate sensing unless the busyness is measured to be longer than $T_{MaxWait}$. Therefore, in QACR-MAC, the scan and handoff procedure is conducted only when there is the definite or highly probable presence of a PU.

Regarding scanning, the TV white space channels can be classified into three categories:

- Empty channel where no activities are detected;
- Active channel where the channel is used by other White-Fi nodes;
- Occupied channel where PU activity is detected.

To reduce the delay caused by scanning, the number of channels to be scanned, N_s, should be reduced. During $S_d[AC]$, channels other than the currently used ones may be assessed to identify if they are empty, active or occupied. This may require increasing $S_d[AC]$, which should be done within the threshold defined for each mode to make sure that it will not significantly affect the QoS metrics and that it complies with regulations. Nodes within the same network should share empty, active and occupied channels that they have identified, using this information to reduce the number of scanned channels. Occupied channels must be excluded from the scanning, and the empty and active channels to be scanned should be limited based on requirements. In the worst case CR users can move to ISM bands to exchange control or data frames if there is no available white space during the given time for scanning channels. This backup channel can be advertised by the AP in its beacon frame or by the initiator of the ad-hoc network.

4.5. Discussion and summary

In this chapter, a sensing strategy that is aware of the QoS requirements of the applications is proposed for White-Fi technology. The proposed 802.11af is mainly designed to assess the spectrum based on the GDB, and more work is required to address the challenges of using a spectrum sensing approach. The current CCA-ED used for implementing Wi-Fi and White-Fi networks cannot perform efficiently in white space because of its limitations and low accuracy. This chapter addresses this research gap in White-Fi. The nature of the random access approach used in White-Fi makes it harder to control the sensing duration and its impact. Existing 802.11af MAC protocols are not designed to handle high-accuracy sensing that has a long duration, so the QACR-MAC is proposed to use a sensing strategy solution that aims to reduce the impact of sensing on MAC protocols and to maintain the QoS mechanism, EDCA. In QACR-MAC, the sensing selection algorithm is used in the first attempt to transmit a data frame and when a collision is deduced. In the case of a collision, the sensing selection algorithm is applied under the constraint of a randomly chosen back-off time, so that it can be exploited with minimal overhead.

The QACR-MAC does not require a CCC channel or centralised coordination, so it is suitable for a wide range of applications. Moreover, QACR-MAC reduces the triggering of costly scan and handoff procedures by involving higher accuracy sensing and the $T_{MaxWait}$ factor. A major concern when relying only on ED sensing is that it will trigger unnecessary handoffs or cause severe interference to PUs; addressing the need for higher sensing accuracy with a suitable mechanism to address the ED limitation is an important step in CR MAC protocol enhancements. The proposed QACR-MAC does not require any significant alteration on the well-known parameters that affect the performance of IEEE 802.11 networks, such as the RTS/CTS mechanism, fragmentation, aggregation, and IEEE 802.11e parameters (see

Section 4.3). In addition, new factors and parameters are involved in fine-tuning the performance of QACR-MAC, in particular, S_d, S_{dMAX}, R_1, R_2, R_3 and $T_{MaxWait}$. For optimised performance, these parameters should be set based on operational conditions, such as the requirements of the sensing methods, the number of the white space channels, and the maximum allowed transmission period in these channels. This research could not find any standard or widely adopted proposed MAC protocol for White-Fi device based only on spectrum sensing. The proposed sensing selection algorithm and QACR-MAC in this chapter comply with current standards and their apparent trends. This work helps to establish a foundation for the future standardisation of enhanced QoS in CR networks, particularly for IEEE 802.11 networks. In the next chapter, the proposed solutions are evaluated and compared with the work described in the literature.

Chapter 5. Simulations and Evaluations

This chapter presents the simulations and evaluation work conducted in this thesis. The purpose of the simulations is to demonstrate the impact of different sensing strategies on QoS. The focus is on evaluating the sensing strategy proposed in Chapter 4, the QoS Awareness MAC protocol (QACR-MAC), and how well it helps to achieve higher QoS in CR networks based on IEEE 802.11 standards. One of the main challenges facing this work was that there are no simulation testbeds available for CR networks based on White-Fi technology, i.e., IEEE 802.11af. Nor are there any widely adopted benchmark metrics to be used as a reference when evaluating the simulation results.

In this chapter the simulation and evaluation efforts in this study are explained, and the results and findings are discussed and evaluated. The chapter starts with a brief introduction in Section 5.1, followed in Section 5.2 by detailed descriptions of how a CR node is implemented with different sensing strategies for studying QoS. Studying the impact of sensing duration on the delay QoS metric under conventional sensing strategies is presented in Section 5.3. The impact of imperfect sensing on the QoS is studied in Section 5.4. Evolution of the proposed select sensing strategy and its improvement, QACR-MAC, are demonstrated in Section 5.5, and the summary and conclusion of the chapter are found in Section 5.6.

5.1. Introduction

Providing the desired QoS in CR networks is an essential requirement if this technology is to succeed. CR networks based on IEEE 802.11 standards, such as White-Fi, are one promising solution, although more work is required to study and improve the QoS in such networks when spectrum sensing is mainly used for spectrum assessment. The main tools used for the simulations in this work are MATLAB 2014a and Riverbed Modeler v18.0.1. MATLAB has been used for modelling Fuzzy Inference Systems (FIS) as explained in Chapter 3 and documented in Appendix A. The Riverbed Modeler (formerly known as OPNET) is used to simulate and evaluate Cognitive Radio (CR) networks under different sensing strategies, operation settings and environments. The Modeler is widely adopted for testing and evaluating communication systems in education and research communities; it is not an open source tool, but it does allow users to customise their implementation codes and models to a reasonable extent. Studying the implementation codes of the standard IEEE 802.11 node in Riverbed Modeler and how to change them to implement different sensing strategies took up a considerable portion of the time available for this work, followed by the time and effort spent on troubleshooting and debugging. While this was at times frustrating, a deep understanding of how 802.11 nodes were implemented in the Modeler and how they might work with various features was gained. One of the advantages of the Modeler is that it offers many features and allows users to gather a wide range of statistics about simulated scenarios. In this chapter, the implementation effort and a substantial amount of the collected statistics and results by simulations are collected, summarised and presented.

5.2. CR node implementations for studying QoS

During this study, CR node with random access (e.g., 802.11af) has not been implemented as a standard model within the well-known wireless simulation tools, i.e., Riverbed Modeler, OMNeT++, NS2, NS3 or NetSim. A CR network based on the 802.22 standard has been implemented in NetSim from TETCOS [208]. As 802.22 nodes do not support random access protocols, implementing a customised wireless node with a random access cognitive radio approach is needed to study different aspects of performance in cognitive radio networks where random access is used. The Riverbed Modeler, v18.0.1, was chosen in this thesis because of its reputation and the ability to provide a wide range of operational features and performance statistics in 802.11-based wireless networks. The Riverbed Modeler wireless suite includes the features of IEEE 802.11, 802.11b, 802.11a, 802.11g, 802.11e, 802.11n High-Throughput (HT), and 802.11p wireless access in vehicular environments (WAVE) standards. These are implemented in various fixed and mobile nodes, including wireless server nodes. Generally, in the Modeler are two wireless local area network (WLAN) nodes: WLAN station (fixed or mobile) and WLAN workstation/server (fixed or mobile). The server node is capable of acting as a wireless access point (AP) and as a server for server–client applications such as email and FTP servers. Also, there are wireless bridge and wireless router nodes that can act as AP. All WLAN nodes share the same MAC layer implementation using the processor ‘wirless_lan_mac’ in the Modeler. WLAN nodes differ based on the included upper layers. The simple WLAN node, WLAN station, has no upper layer protocols, but it has source and sink processors to generate packets for transmission and destroy received packets respectively (see Figure 5.1 a). This kind of node is used for simulating the MAC layer features without considering the upper layer settings, including the lack of routing capability. The advanced WLAN node; that is, the workstation/server node, has upper layers implemented that allow this kind of node to support various settings for running applications like ad-hoc routing parameters and other upper layer protocols (see Figure 5.1 b).

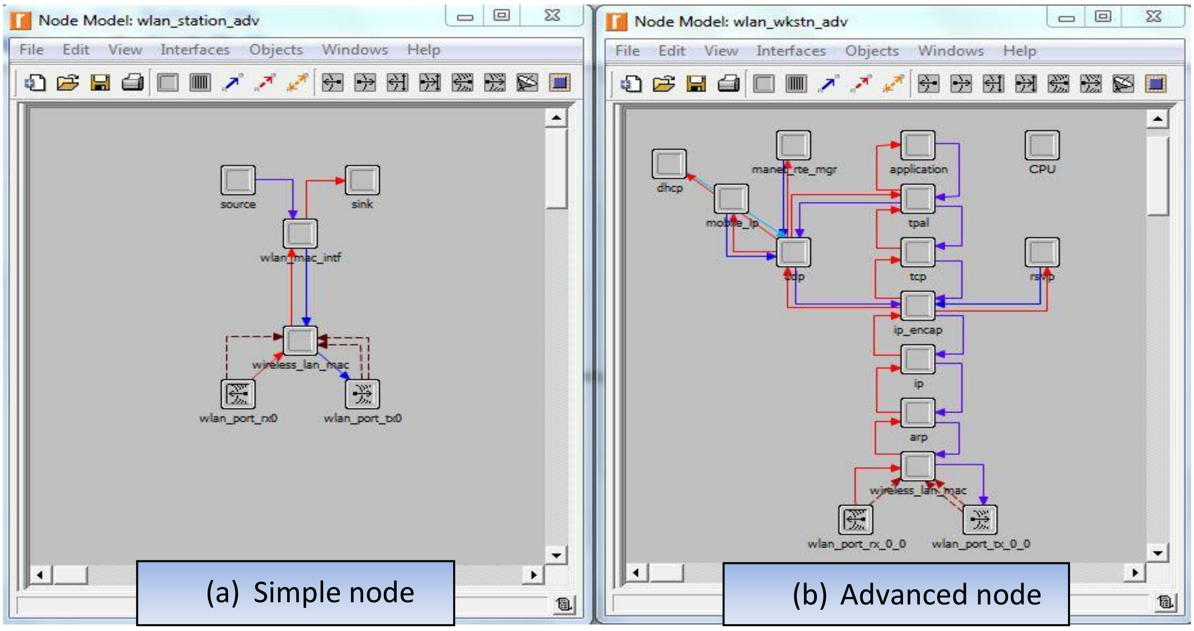


Figure 5.1 Wireless 802.11 based nodes: (a) simple node, (b) advanced node

The Riverbed Implementation of WLAN MAC layer was studied to find a way to customise it to act as a CR node for this study. Although the Riverbed Modeler is not an open source software, it allows researchers, to some extent, to customise and implement their own nodes using a range of Riverbed functions library and C/C++ code.

5.2.1. 802.11 MAC layer customisation for sensing

The IEEE 802.11 protocol for WLAN nodes is already based on sensing. The sensing function in the IEEE 802.11-based nodes of Riverbed Modeler is implemented under the assumptions that there is no sensing duration and there is perfect sensing. These assumptions can be expressed as $S_d = 0$, $P_d = 1$ and $P_f = 0$, where S_d is the sensing duration, P_d is the probability of detection and P_f is the probability of false alarm. Essentially, the sensing is based on calculating the reception power against a pre-set sensing threshold. It is not conducted during the network allocation vector (NAV) when virtual sensing is used. For the CR node, sensing should have different possible durations, P_d, P_f, and strategies. It is required to add sensing function to the standard MAC layer model in the Modeler that includes such parameters, to study their impact on performance in a random access approach. The 'wirless_lan_mac' process has a parent process model called 'wlan_dispatch', as shown in Figure 5.2. This parent process can call one of the child processes 'wlan_mac' or 'wlan_mac_hcf' based on physical characteristic settings in the node attributes. The 'wlan_mac_hcf' process is used when the node is set to support the hybrid coordination function (HCF), specified in the IEEE 802.11e standard as a mandatory feature [196]. The 'wlan_mac' process is used for nodes configured with legacy 802.11 MAC protocols which will not be implemented in future CR capability nodes, and as the customisation is done on the 'wlan_mac_hcf' process for CR sensing, the focus of this work is on the 'wlan_mac_hcf' process.

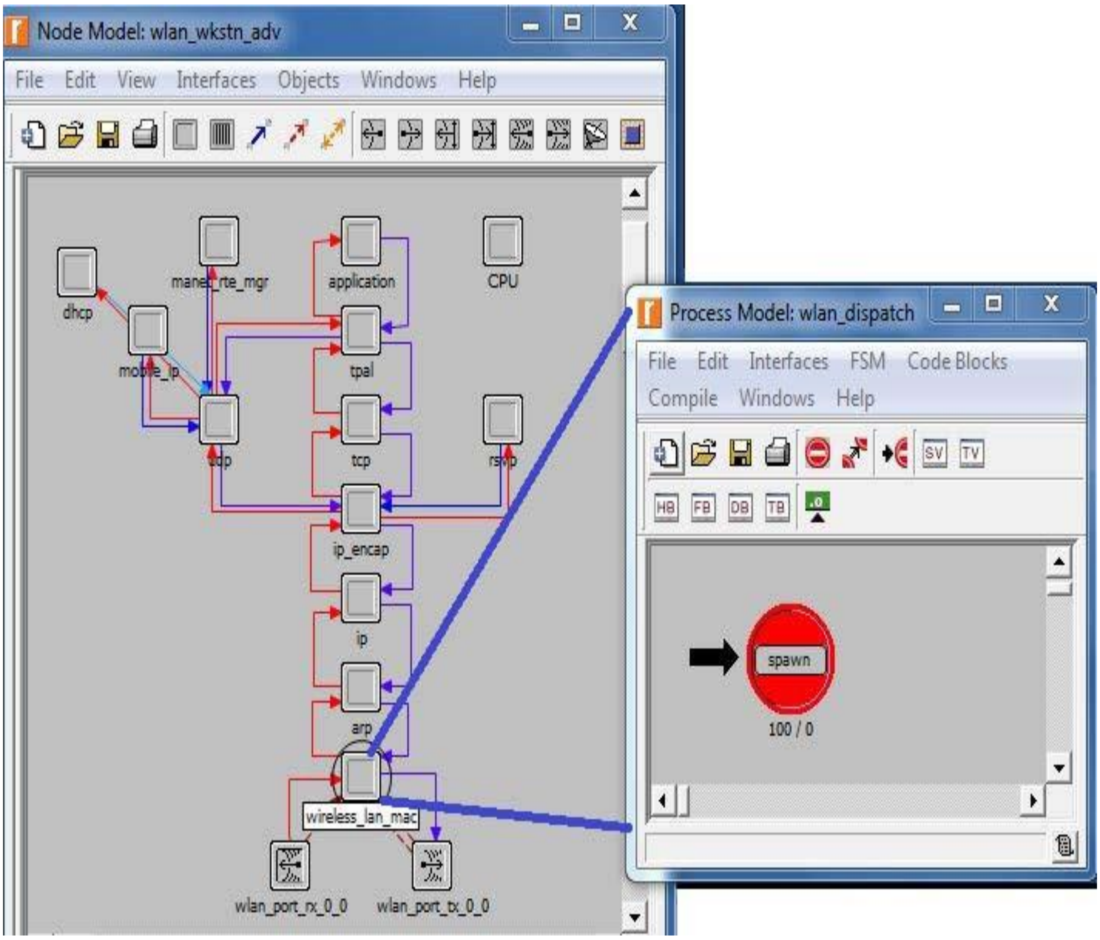


Figure 5.2 The parent wlan_dispatch of the wireless_lan_mac process model

In this work, a customised sensing function was implemented so when to sense, and the duration and accuracy of sensing parameters, could be changed according to the study requirements. The accuracy of sensing is represented by the P_d and P_f parameters, usually measured by distinguishing between busy and idle states of the channel regardless of which signal type is occupying the channel in the busy state. The unforced state process Sense, or Sense state, was created for the sensing function within the child process model 'wlan_mac_hcf'. The main functions of the Sense state are illustrated in Figure 5.3 (see Appendix B for the entire code used in the Sense state). Then several customised 'wlan_mac_hcf' process models were implemented with diverse sensing strategies and a different name given to each model, starting with the original name 'wlan_mac_hcf' and ending with different suffixes '_cr_*X*', (see Table 5.1). We can change the sensing strategy by changing the process model of the 'Wireless_lan_mac' to one of the customised models, shown in Figure 5.4, which were created in this study. The sensing strategy term refers to the adopted approach of conducting spectrum sensing, in terms of when to sense, sensing duration, which sensing method to use, and the action to be taken based on the sensing outcome. A sensing strategy is implemented according to how the Sense state will be configured and how it will interact with other states in the process model 'wlan_mac_hcf_cr_*X*'. For example, in the customised process model 'wlan_mac_hcf_cr_noIFSPCH', the Sense state is connected to the other three states as shown in Figure 5.5. The Sense state will be conducted after the IFS and BACKOFF state, and based on the sensing outcome the Transmit state or Scan state will be conducted. The red colour is used for an unforced state and green for a forced state. The main difference between unforced and forced is execution timing. An unforced state process affects simulated time and releases control to the simulation kernel between its enter and exit code executives, while a forced state invokes enter and exit executives without releasing control to the simulation kernel or affecting simulated time. Transition lines between states indicate the flow of executing states within a process under defined conditions. In this work, the code and design customisations are done to implement CR nodes with different sensing settings and strategies are documented in Appendix B. The comparison of the standard process model 'wlan_mac_hcf' and the customised process model 'wlan_mac_hcf_cr_noIFS' is shown in Figure 5.6.

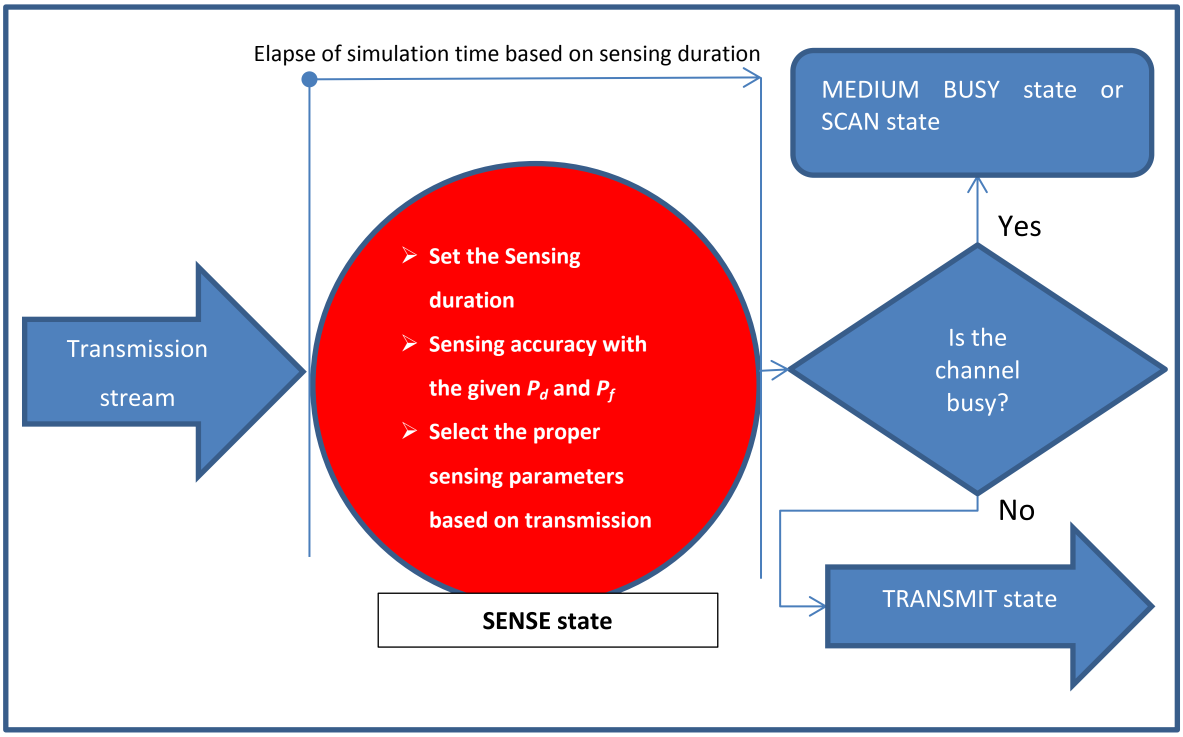


Figure 5.3 Sense state functions

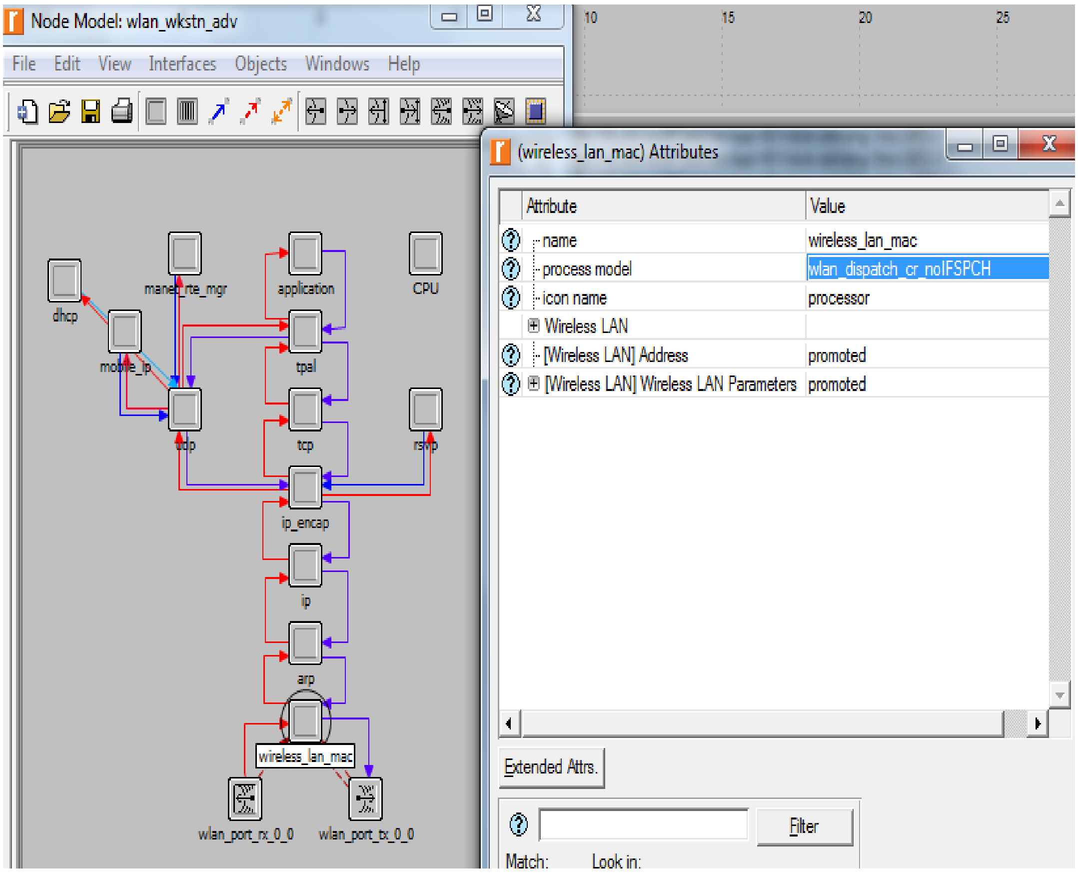


Figure 5.4 Changing the node's MAC protocol to a customised process model

Table 5.1 Different MAC model implementation based on sensing strategy.

MAC process model (Sensing strategy name)	Child process model for HCF	Sensing strategy summary
wlan_dispatch_cr_adv **(Extreme sensing strategy)**	wlan_mac_hcf_cr_adv	Sensing before sending any frame.
wlan_dispatch_cr_noIFS **(Fixed sensing strategy** with perfect sensing**)**	wlan_mac_hcf_cr_noIFS	Sensing before sending any frame except response frames.
wlan_dispatch_cr_noIFSP **(Fixed sensing strategy** with imperfect sensing**)**	wlan_mac_hcf_cr_noIFSP	Adding accuracy factors; i.e., P_d and P_f, to fixed sensing strategy.
wlan_dispatch_ cr_noIFSPCH **(Fixed sensing strategy** with imperfect sensing**)**	wlan_mac_hcf_cr_noIFSPCH	Adding changing operation channel to fixed sensing strategy.
wlan_dispatch_cr_noIFSPS **(Select sensing strategy)**	wlan_mac_hcf_cr_noIFSPS	Conducting selection approach before sending any frame except response frames.
wlan_dispatch_cr_noIFSPS4 **(QACR-MAC sensing strategy)**	wlan_mac_hcf_cr_noIFSPS4	Conducting selection approach only for the first transmission attempt of a frame.

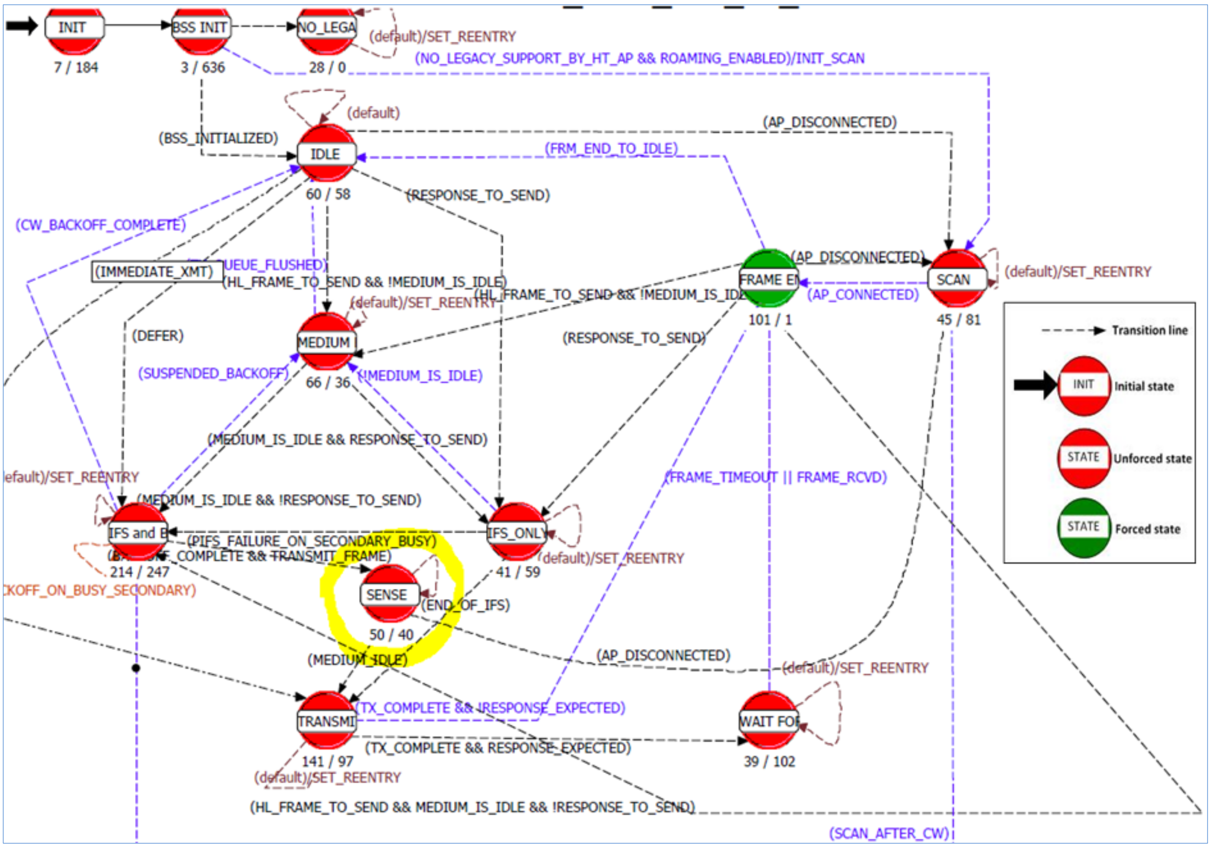


Figure 5.5 Process model 'wlan_mac_hcf_cr_noIFSCH' with the added Sense state (Highlighted)

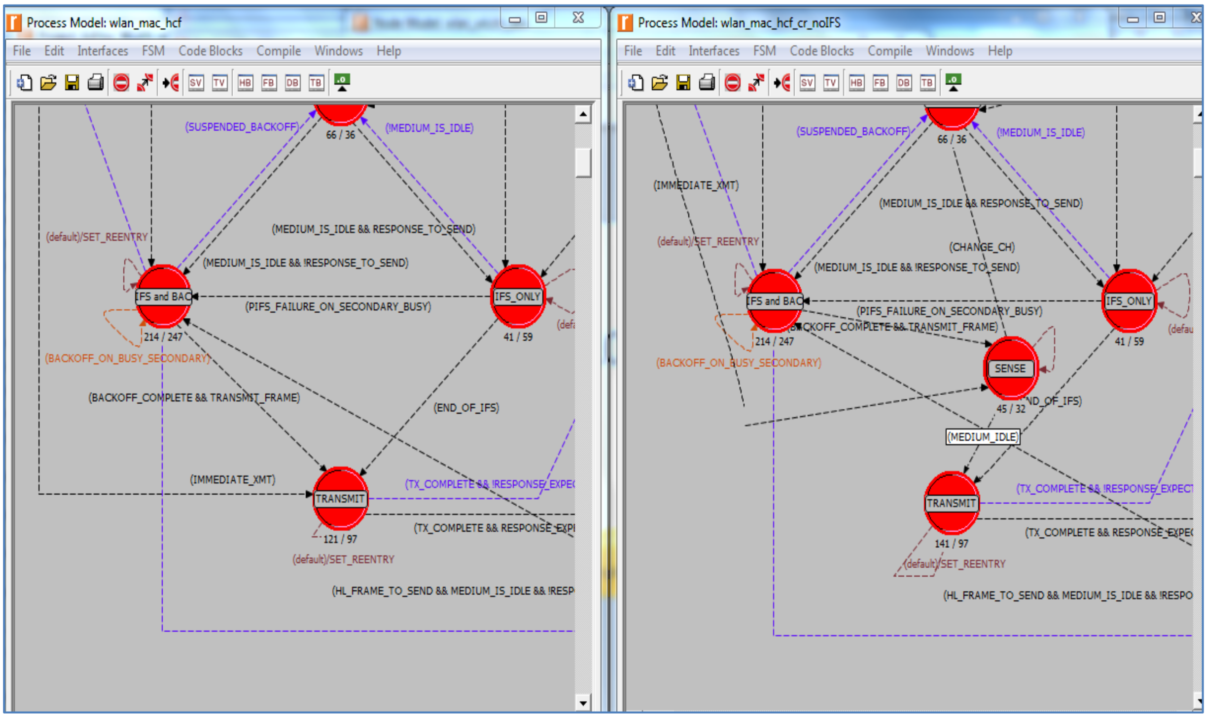


Figure 5.6 Comparing the standard process model 'wlan_mac_hcf' and the customised model 'wlan_mac_hcf_cr _noIFS'

The sensing methods differ in accuracy, based essentially on these parameters: the sensing duration, P_d, P_f, and the ability to identify a PU signal from other signals in the channel. Simulating different sensing methods can be conducted by setting those parameters to reflect the given sensing method. As proposed in the previous two chapters, selecting the proper sensing method is restricted by factors such as available prior information about the PU signal and the capability of the CR device. In this simulation work, the assumed values of these factors discussed in Section 4.4.1 were also used.

5.2.2. Features and limitations of MAC layer implementation

The CR customised nodes that were implemented in this study are subject to the features and limitations of the standard wireless 802.11-based node in Riverbed Modeler 18.0.1. In this section, the features and limitations related to the simulation work in this thesis are highlighted in the following points [209]:

- **HCF** is implemented to support enhanced distributed channel access (EDCA) with the following characteristics:
    - Four access categories (ACs) for prioritised contention-based access: voice, video, best effort, and background;
    - EDCA parameter set distribution by an AP;
    - Transmission opportunity (TXOP) frame bursting is supported when HCF capability or 802.11n is enabled; an STA can transmit multiple frames in a TXOP if the value of the TXOP limit for the access category is greater than zero. Similarly, other frames in the TXOP reserve the medium for the remaining duration of the TXOP.
- **RTS-CTS mechanism:** a reliable data transmission is supported via threshold RTS, based on data frame size.
- **Fragmentation and reassembly:** optional data frame fragmentation is supported, based on the size of the data packet received from the higher layer. The fragments are reassembled at the destination station.
- **MAC-level acknowledgements:** the model supports normal ACK, Block-ACK, and no-ACK MAC-level acknowledgement mechanisms. A MAC acknowledgement requirement can be configured separately for each traffic category. Block

acknowledgement (B-ACK) support includes:

- B-ACK agreement set-up with the exchange of action management frames;
- Immediate B-ACK and delayed B-ACK;
- B-ACK support for roaming;
- Inactivity detection and B-ACK tear-down.

- MAC and physical layer features:
  - Aggregated-MAC service data unit (A-MSDU) and A-MPDU;
  - No ACK, immediate block ACK and delayed block ACK are supported with unicast A-MPDU frames;
  - APs can transmit broadcast A-MPDU frames;
  - Implicit block ACK requests will be used with Immediate block ACK agreement;
  - A block ACK agreement between two HT-STAs uses the compressed B-ACK frame;
  - Data MPDUs belonging to a block ACK agreement between two HT-STAs will not be fragmented;
  - FHSS, IR, DSSS, OFDM, extended rate PHY-OFDM (802.11g), and high-throughput PHY (802.11n) are supported;
  - Short guard intervals, which allow higher data rates;
  - Multiple-input and multiple-output (MIMO) capability, which allows higher data rates (however, the physical layer details of MIMO are not currently implemented);
  - Carrier sensing in the secondary channel for 40MHz transmissions.
- **Coexistence Support:** interference between WLAN and ZigBee networks operating in overlapping bands is supported.

The following limitations are some of the MAC protocol features that are not supported in Riverbed 18.0.1 Modeler [209]:

- Management frames are not modelled, with the exception of beacons and block-ACK-related action management frames. Transmission of periodic beacons in ad-hoc mode is not modelled.

- Power save mode is not modelled.
- Authentication and security procedures are not modelled.
- The transmission rate used for data transmission is static throughout the simulation.
- Roaming and point coordination function (PCF) are implemented separately, but cannot be used together.

All the above features and limitations are reported in the Riverbed Modeler documentation. New standards such as 802.11ac and 802.11af are not modelled yet, even in the last updated version of Riverbed Modeler, 18.6.0.

5.3. Sensing duration impact on the delay QoS metric

The sensing strategy is mainly influenced by the MAC protocol used, but other factors may also influence it. In Section 2.5, several sensing strategies are identified and discussed. For contention-based MAC protocols, there is no widely accepted sensing strategy. In this section, two basic approaches based on fixed sensing duration along the operation were compared. The first strategy uses an extreme sensing frequency, where sensing is conducted before sending any frame. The second strategy is less frequent than the first; it discards sensing in respond frames, which is similar to the CR-CSMA protocol proposed in [140]. The CR-CSMA is used as a punch-mark in several studies to evaluate their improvements, as discussed in Section 2.5. The CR-CSMA was proposed based on a fixed sensing method with Energy Detection (ED). It has been noted many times throughout this thesis that ED is not sufficient for sensing in the CR environment, and more advanced methods with longer sensing duration are needed.

The simulations in this section were conducted to illustrate the impact of sensing duration on the QoS of different applications. Different sensing strategies were used in these scenarios, but all under the assumption that P_d = 1 and P_f = 0; that is, of perfect sensing. The PU is not present in this scenario, no other SU systems share the operational channel, and the network nodes are within range of each other in a high SNR environment. This network topology is designed to avoid the possibility of a hidden terminal problem, because this section concentrates on illustrating the impact of sensing duration and frequency in 802.11-

based networks, on various applications running on CR nodes. The simulations in this section were conducted under the given parameter settings shown in Table 5.2.

Table 5.2 Wireless node settings.

Parameter	Value
Physical characteristics	HT PHY 5.0 GHz
Data rate	26 Mbps (base)/ 240 Mbps (max)
Transmit power	0.005 W
Buffer size	1024000 bits
EDCA parameters: (default settings)	
Voice	CWmin= (PHY CWmin + 1) / 4-1
	CWmax= (PHY CWmin + 1) / 2-1
	AIFSN = 2
	TXOP = One MSDU
Video	CWmin= (PHY CWmin + 1) / 2-1
	CWmax= PHY CWmin
	AIFSN = 2
	TXOP = One MSDU
Best Effort	CWmin=PHY CWmin
	CWmin=PHY CWmax
	AIFSIN=3
	TXOP = One MSDU
Background	CWmin=PHY CWmin
	CWmax=PHY CWmax
	AIFSIN = 7
	TXOP = One MSDU
Frame aggregation parameters:	
Maximum transmitter aggregated MAC service data unit (A-MSDU) size	3839 bytes
Maximum acceptable A-MSDU size	8191 bytes
Minimum acceptable MPDU start spacing	No Restriction

The RTS/CTS option was disabled as it is not efficient in these scenarios as only a few nodes were involved, with a topology in which the hidden node problem is avoided. The IEEE 802.11e was supported in all scenarios with the default settings illustrated in Table 5.2. Simulations were conducted under different sensing durations and for different application categories. The main three types of application considered were voice traffic, video conferencing traffic, and heavy email traffic. The resulting values were captured under 'Bucket mode' with a sample mean of 100 values per a result statistic. In Bucket mode, the data is collected at all of the points over the time interval or sample count into a 'data bucket', and a result is generated from each bucket.

There were three types of delay collected in these simulations: wireless delay, wireless delay per AC, and media access delay. The wireless delay, simply called 'delay' in this work, represents the end-to-end delay of all the data packets that are successfully received by the MAC layer and forwarded to the higher layer in a node. This delay includes queuing and medium access delays at the source node MAC layer, reception of all the fragments individually, and the relay of the frame via the AP in the infrastructure mode. This type of delay is also collected separately for frames belong to different HCF ACs. The media access delay is the sum of delays, including queuing and contention delays of all frames transmitted via the MAC layer. The media access delay is calculated as the duration between the time when the frame is placed in the transmission queue until the time when the frame is sent to the physical layer for the first time. In other words, the media access delay is the time of processing a frame at the MAC layer. This frame is created either from a higher layer data packet when it arrives at the MAC layer or from a protocol frame created at the MAC layer.

For voice application, voice traffic was generated as IPv4 unicast traffic flows between the nodes. A sample of the traffic flow generated between two nodes is shown in Figure 5.7. For generating email and video conferencing traffic, the standard application modelling, using application definition and profile definition nodes, were used to model these types of traffic.

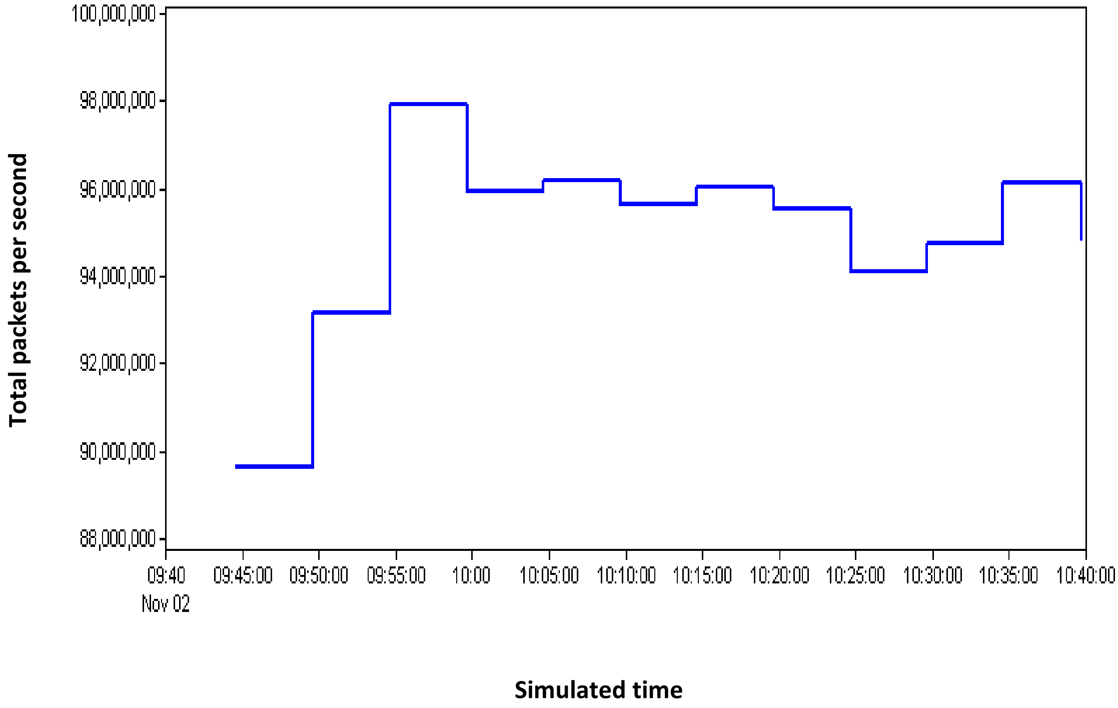


Figure 5.7 Total voice traffic generated between two nodes

5.3.1. Two fixed sensing strategies in an ad-hoc network with voice traffic

Two simulation scenarios were implemented to analyse the QoS impact of sensing duration for two sensing strategies: extreme sensing strategy and fixed sensing strategy, as described in the first two rows of Table 5.1). In the first scenario, an ad-hoc network of two nodes using 'wlan_dispatch_cr_adv' is simulated with voice traffic generated between them. A snapshot of the two nodes in ad-hoc mode is shown in Figure 5.8. One-way voice traffic is shown in Figure 5.7, and the same traffic is generated for the opposite direction. The two nodes are customised to conduct the sensing before sending any frame, which represents an extreme approach to protecting the PU.

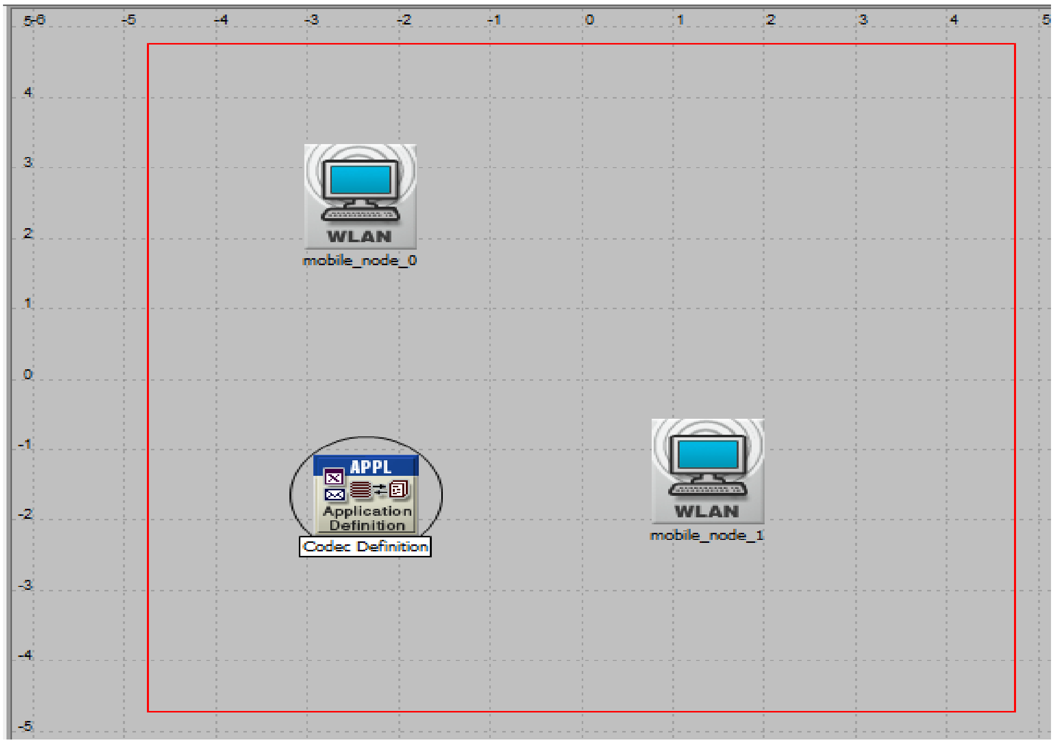


Figure 5.8 Two-node ad-hoc network for comparing sensing strategies

The simulations were run under sensing durations of 1 ms, 5 ms, 10 ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300 ms and 350 ms. The MAC protocol cannot handle transmission between two nodes when the sensing duration is 450 ms or more. In Figure 5.9, the average traffic delays under different sensing durations were measured over half an hour of network operation. The simulation was also run when the nodes operated without CR capability where the sensing duration was neglected: that is equal 0 ms. From Figure 5.9, the average delay exceeds two seconds when the sensing duration is 250 ms or more; it is over one second for sensing durations of 150 ms and 250 ms. The average delay is under one second for a sensing duration of 100 ms or less. Figure 5.10 focuses on the difference between the average delay for the 1 ms duration and when the sensing duration is neglected. The average delay is around 9 ms when the sensing duration is 1 ms, while the average delay is around 0.04 ms when sensing is neglected.

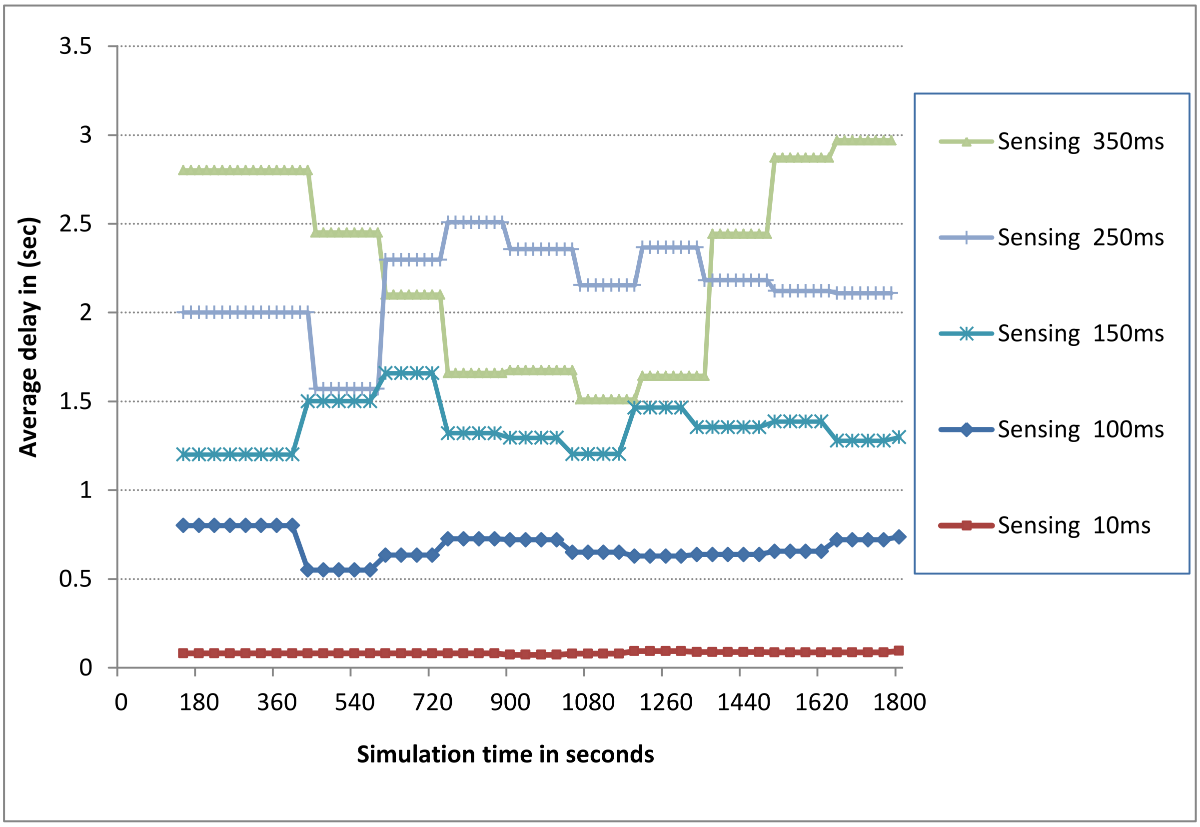


Figure 5.9 Average delays for different sensing durations for extreme sensing strategy (ad-hoc network of two nodes with voice traffic)

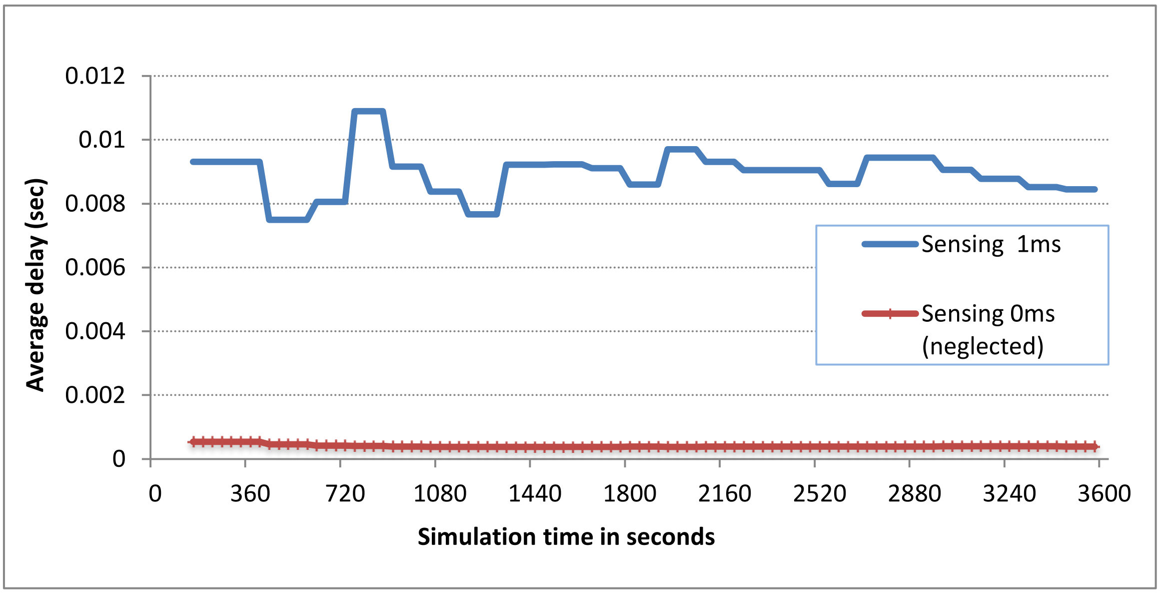


Figure 5.10 Comparing between the average delays of sensing durations 1 ms and 0 ms (neglected) for extreme sensing strategy (ad-hoc network of two nodes with voice traffic)

In the second scenario, the two nodes performed sensing before sending any frame except a response frame: that is, a frame after an SIFS. The sensing frequency was reduced so sensing methods requiring longer duration can be used in this strategy. Although the PU protection level will be affected, the response frames are usually small, so their degree of effect on the detection of the PU signal can be tolerated. Moreover, using high-accuracy sensing methods with a longer sensing duration helps in improving PU protection. Several simulations were run under different sensing durations: 1 ms, 5ms, 10 ms, 50 ms, 100 ms, 150 ms, 200 ms, 250 ms, 300ms, 350 ms and 400 ms.

Figure 5.11 shows samples of the average delays under these various sensing durations for the voice traffic shown in Figure 5.7. The average delay, in this case, fluctuates less than in the first scenario, when sensing is performed for all frames including response frames. In the one hour of the simulated network operation, the average delay gradually decreased until it reached a plateau state. It can be seen in Figure 5.11 that the delay is reduced compared to the first scenario, even when the sensing duration is 400 ms; the average delay is less than 1.6 seconds. The average delay is less than one second for a sensing duration of 250 ms or less. For an average delay of less than 0.2 seconds, the sensing duration is less than 50 ms.

The average media access delay caused on the MAC layer because of the sensing duration 1 ms is shown in Figure 5.12. It can be seen that the average media access delay is around 1.6 ms for a 1 ms sensing duration. If we compare that with the average media access delay in the first scenario, shown in Figure 5.13, an extra overhead of 3 ms is added when sensing is conducted before all frames. This overhead is doubled when the sensing duration is doubled.

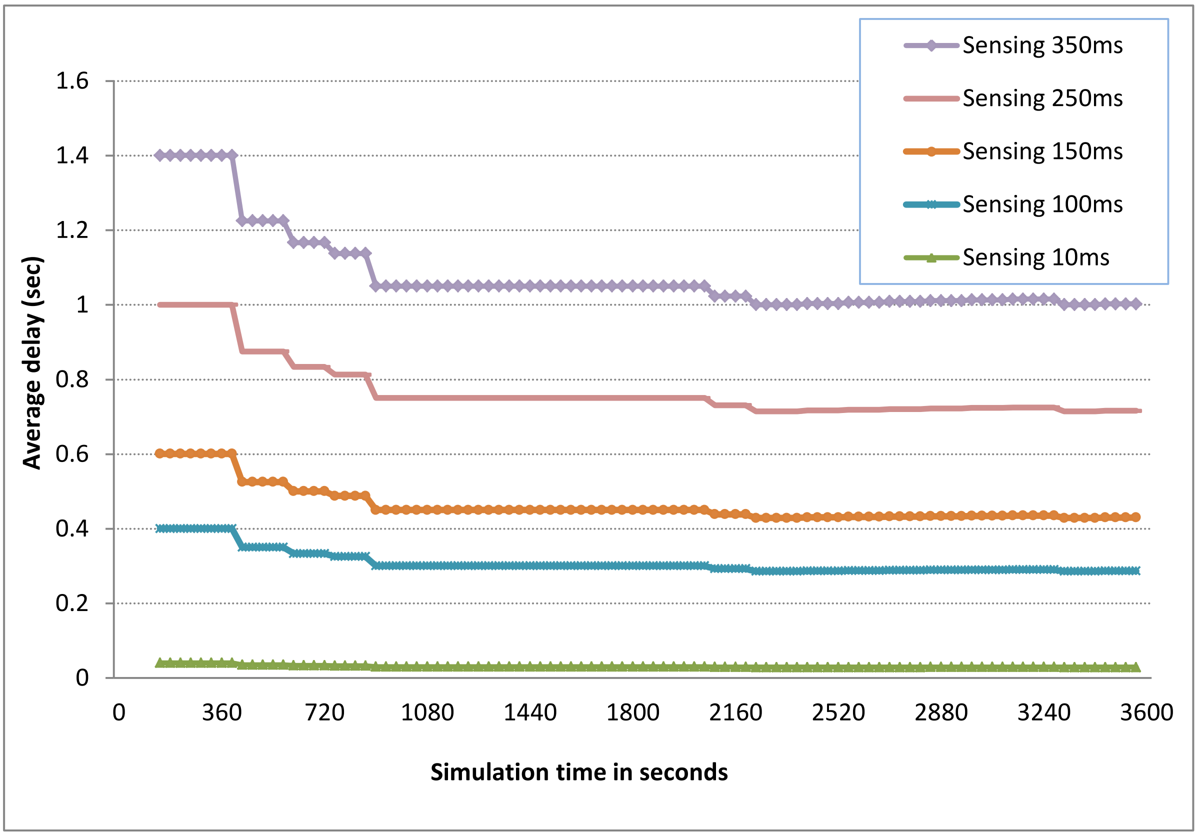


Figure 5.11 Average time delays for different sensing durations for fixed sensing strategy (ad-hoc network of two nodes with voice traffic)

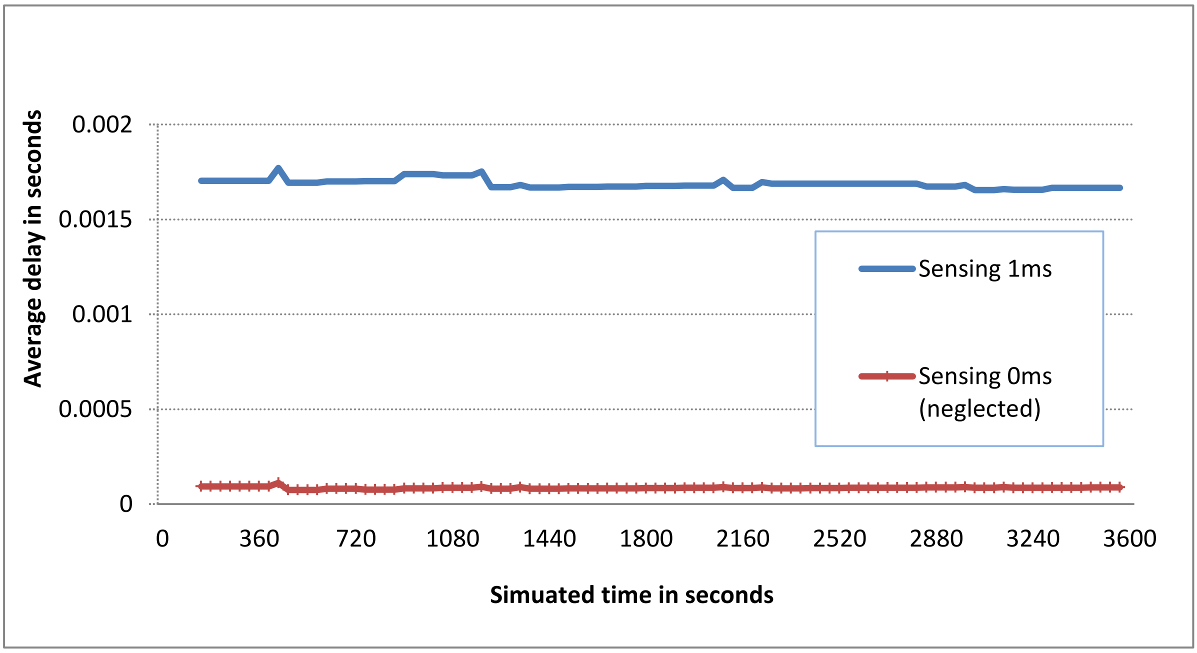


Figure 5.12 Average media access delay comparison between the sensing durations 1 ms and 0 ms (neglected) for fixed sensing strategy (ad-hoc network of two nodes with voice traffic)

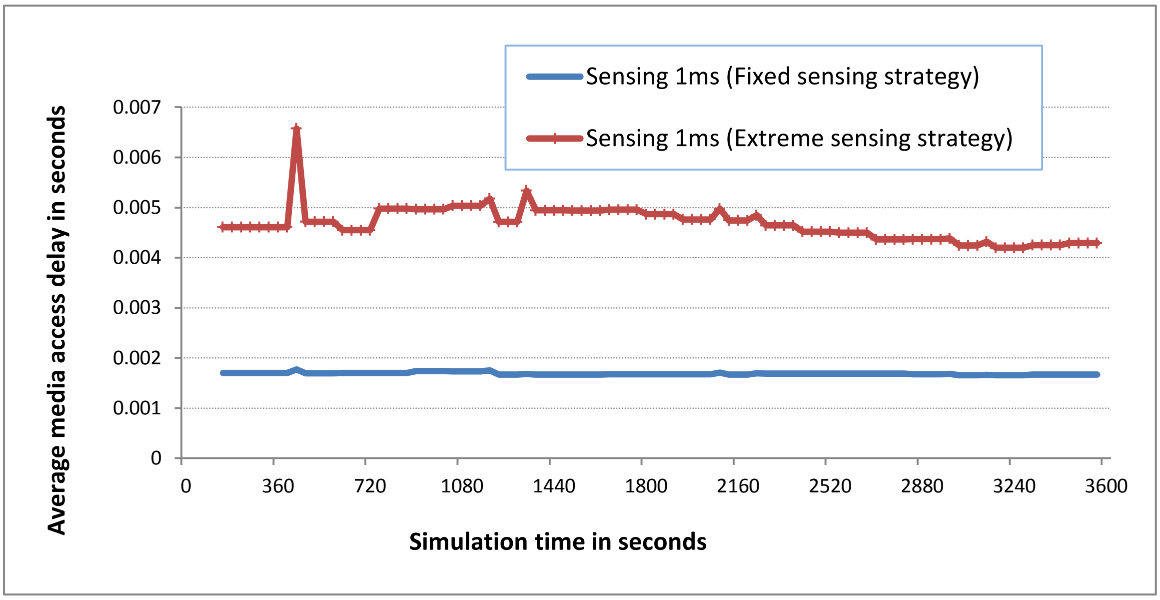


Figure 5.13 Average media access delay comparison between extreme sensing and fixed sensing strategies for 1 ms sensing duration (ad-hoc network of two nodes with voice traffic)

5.3.2. Impact of sensing duration on different types of applications

In this subsection, the effect of sensing duration was analysed for various applications. Three scenarios are implemented for that purpose: voice scenario, email scenario and video scenario. The sensing was conducted for all frames except response frames in all scenarios. For voice scenario, a four-node ad-hoc network was used, as shown in Figure 5.14. In addition, IPv4 unicast voice traffic was generated between all the four nodes.

Figure 5.14 Four nodes ad-hoc network (scale in meters)

Several simulations were conducted for different sensing durations from 1 ms to 500 ms. Simulations were also run when the sensing duration was neglected: that is, 0 ms. According to Figure 5.15, the average delay for each of the sensing duration starts to increase sharply from the beginning until the 900 seconds of simulated time. It becomes more stable afterward for all sensing durations. It can be seen in Figure 5.16 that the average delays reach almost the same steady-state value for both the two-node and four-node networks for the same sensing durations.

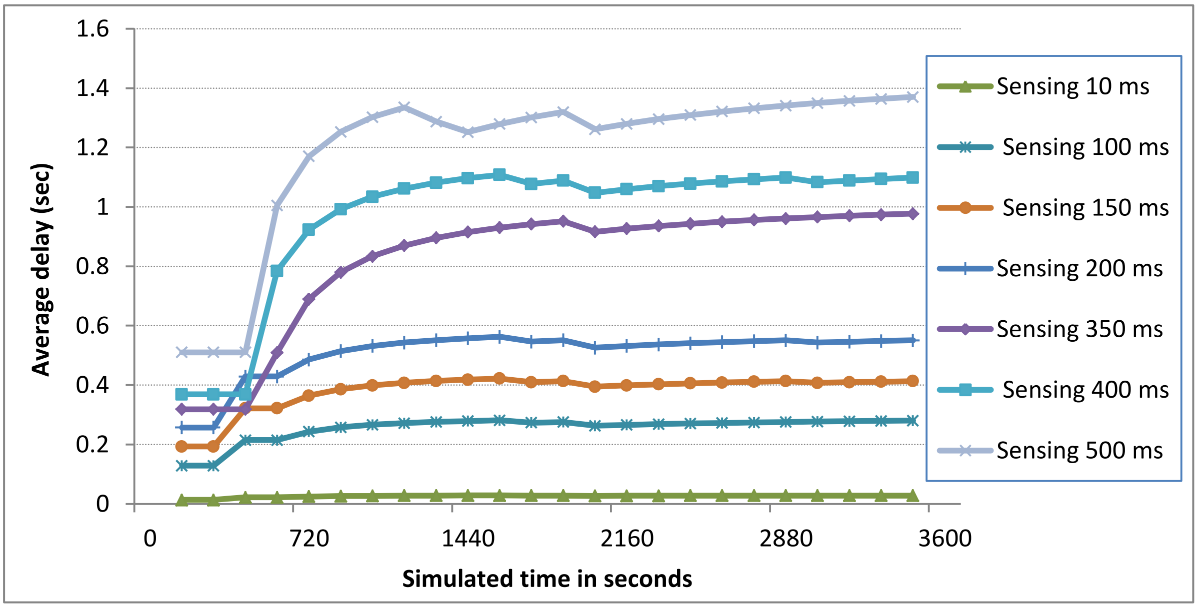


Figure 5.15 Average delays for different sensing durations for fixed sensing strategy (ad-hoc network of four nodes with voice traffic)

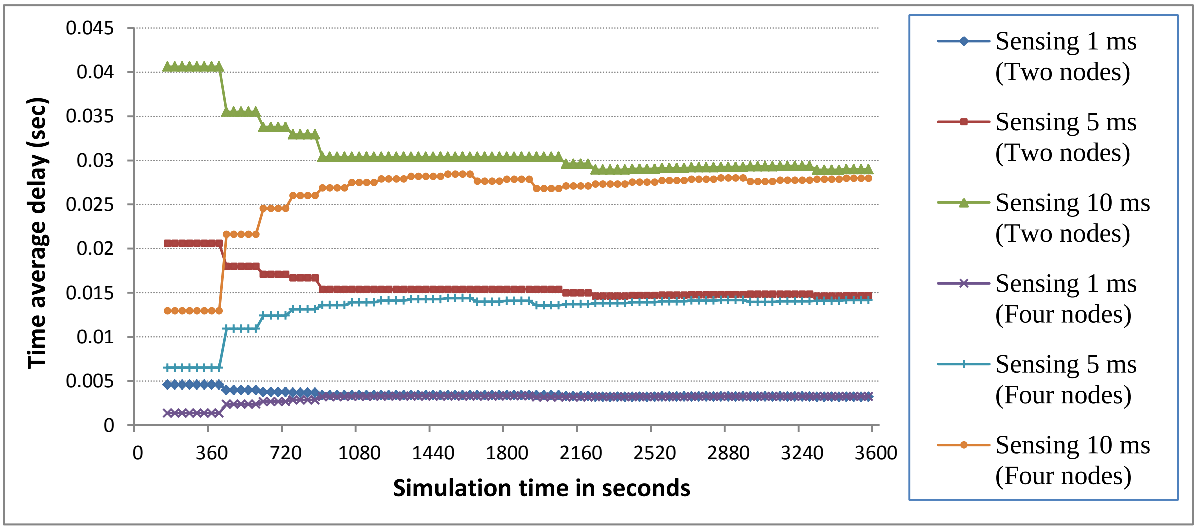


Figure 5.16 Average delay comparison between two- and four-node networks for fixed sensing strategy (ad-hoc network with voice traffic)

In the email scenario, an email server was added to simulate heavy email traffic between the server, node_2, and other four nodes, as shown in Figure 5.17. Instead of operating as an ad-hoc network, as in the previous scenario, the network in this scenario was operated in infrastructure mode with the server also acting as an AP. The email application traffic was configured using the application definition node, node_1, and the profile definition node, node_0, which are provided in Riverbed Modeler for this purpose. Several simulations were run under different sensing time durations, from 1 ms to 500 ms, the simulated operation time of the network was an hour for each. The simulation results of the average delay are shown in Figure 5.18. According to the results, the maximum measured average delay is 0.6 seconds when the sensing duration is 500 ms. The average delay is between 0.3 and 0.1 seconds for sensing durations between 250 ms and 100 ms, while it is less than 0.1 seconds for sensing durations less than 50 ms. These results show that the delay is less for heavy email traffic than for voice traffic.

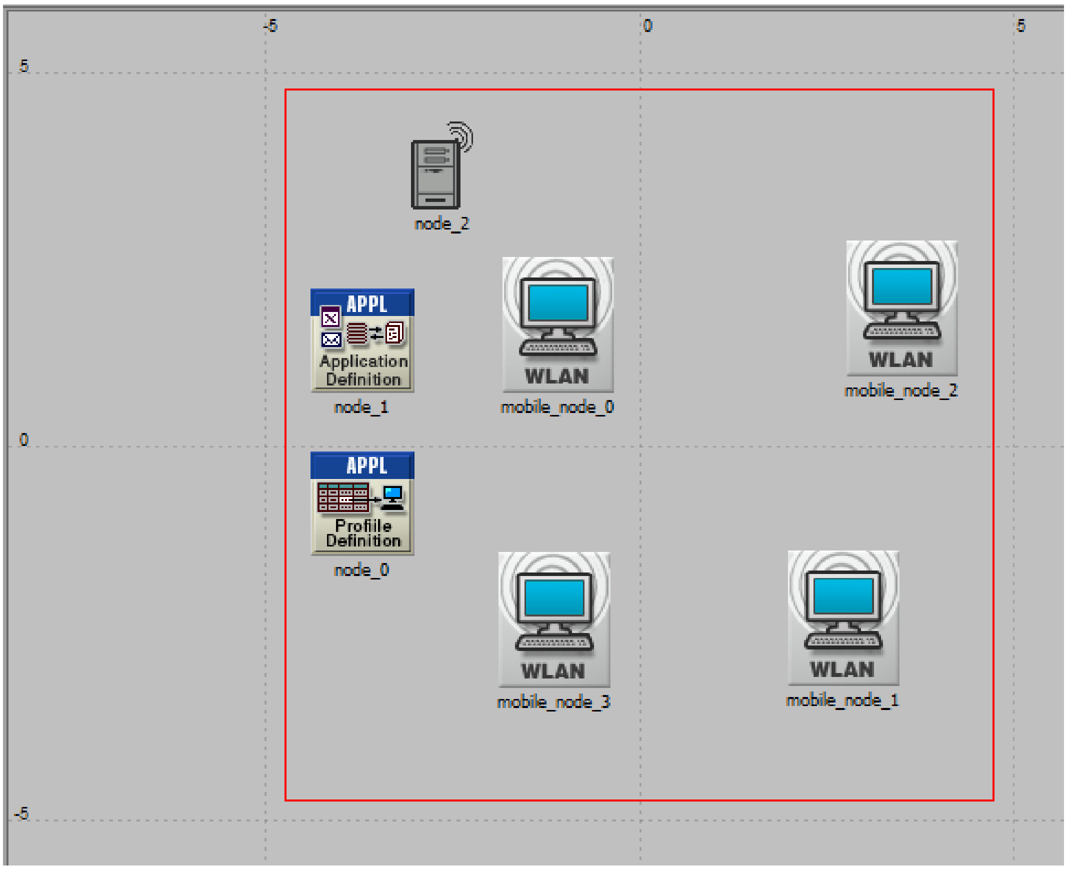


Figure 5.17 Five-node infrastructure WLAN network (scale in meters)

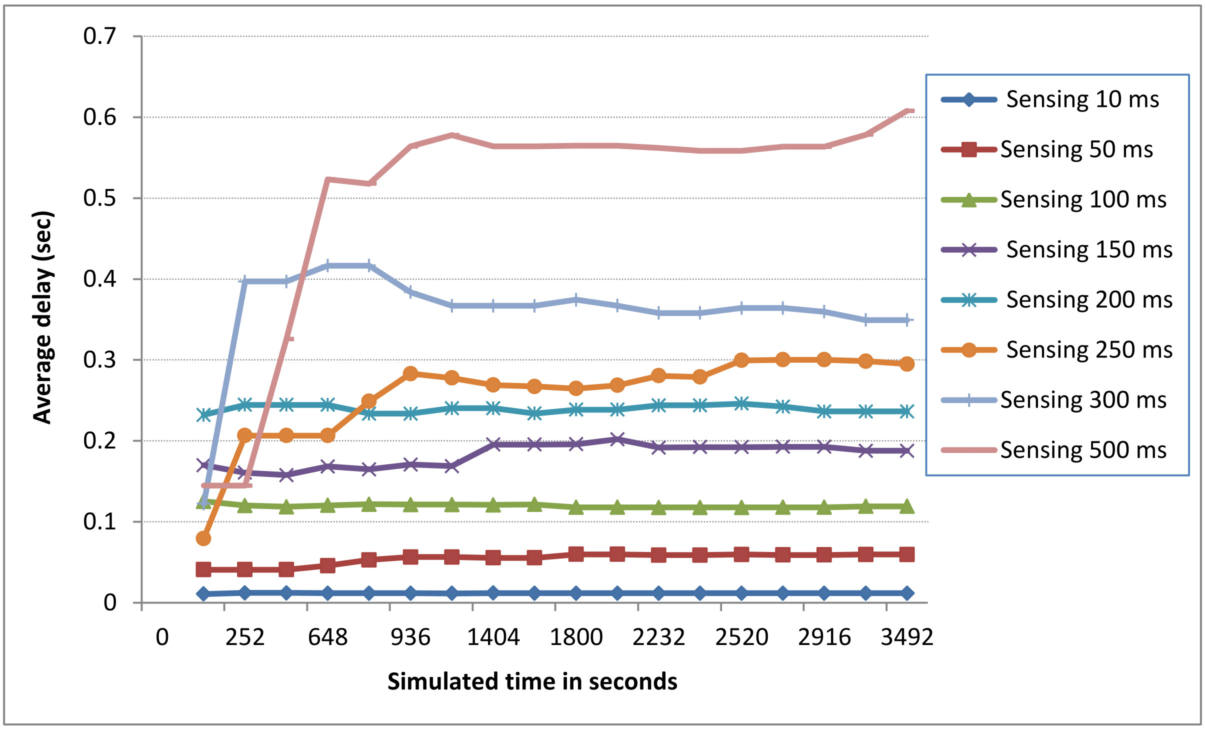


Figure 5.18 Average delays for different sensing durations for fixed sensing strategy (infrastructure network with heavy email traffic)

In the video application scenario, the four nodes used for the exchanging video conferencing application traffic through an AP node. The traffic was configured with a high-resolution video with a frame rate of 15 frames/second and a frame size of 240x128 pixels. The simulations were conducted for different sensing durations from 1 ms to 500 ms. The average delay in this scenario did not exceed 0.1 seconds even when the sensing duration was 500 ms, as shown in Figure 5.19. It can be seen that the 350 ms and 400 ms sensing durations cause the highest average delay, over 0.08 second, and is greater than the average delay for the cases of 450 ms and 500 ms sensing durations. The average delay in this scenario does not follow a proportional relationship between the average delay and sensing duration. An optimal sensing duration could be achieved for a long sensing duration. In Figure 5.20, the trend-line of the variants of the average delay, caused by the sensing time durations, shows that 100 ms, 300 ms, and 500 ms sensing durations may lead to optimal potential values of the average delay.

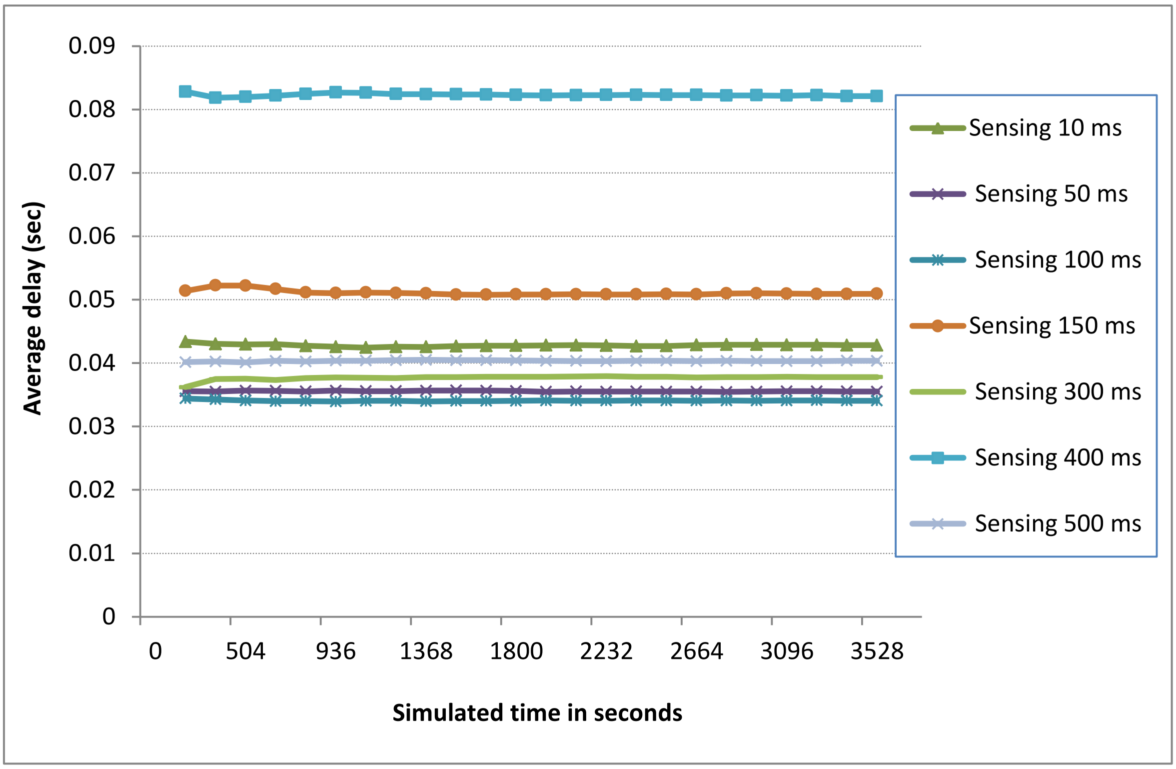


Figure 5.19 Average delays for different sensing durations for fixed sensing strategy (infrastructure network with video conferencing traffic)

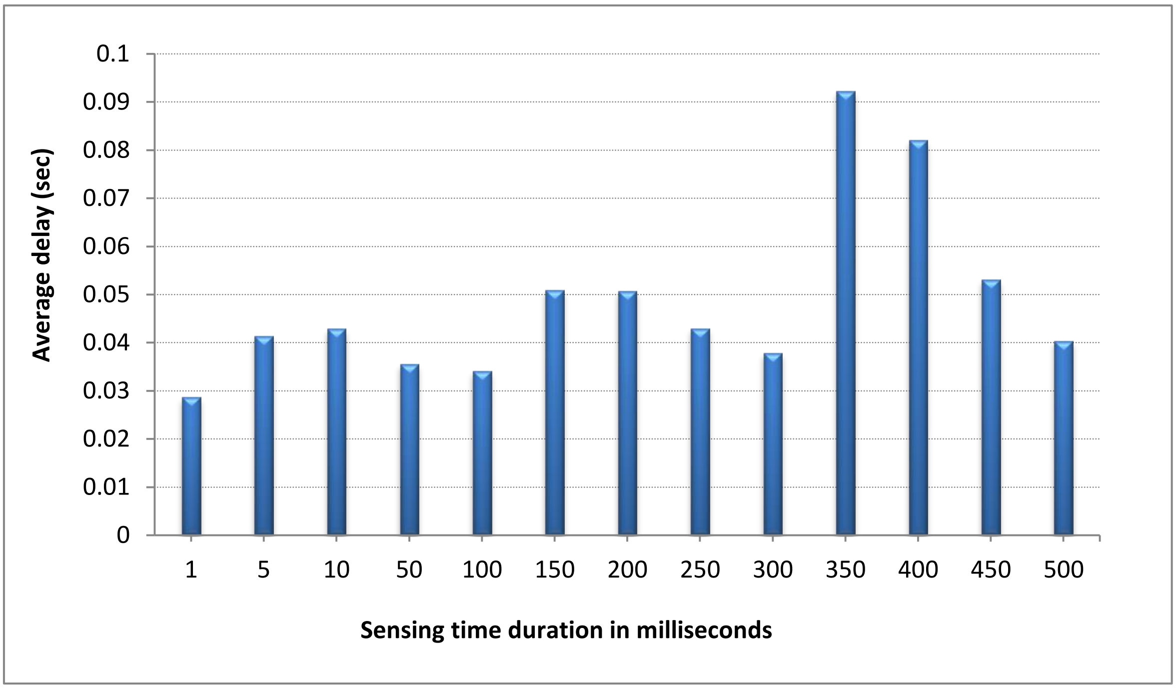


Figure 5.20 Average delay comparison between different sensing durations for fixed sensing strategy (infrastructure network with video conferencing traffic)

5.3.3. Observations and comparisons of sensing duration impact on QoS

When sensing is conducted more frequently, as in the extreme sensing strategy, the average delay is affected significantly more than the fixed sensing strategy, where the sensing frequency is reduced. Figure 5.13 shows that the main source of increasing delay is, in the first scenario, because of the MAC layer processing; that is, media access delay. Although conducting sensing before sending any frame provides more protection to PU signals, the resulting delay has a significant impact on QoS, particularly in applications sensitive to delay like voice applications. Excluding sensing before sending response frames reduces such an impact and improves the achieved QoS in White-Fi devices, the result of comparing fixed and extreme sensing strategies. As far as PU protection is concerned, the response frames are small and the possibility of their interfering with PU signals is insignificant. The comparison between the three traffic types, voice, email and video, clearly illustrates that voice traffic experiences the highest average delay, as shown in Figure 5.21: compared with the others, voice traffic is more sensitive to the sensing duration. The average delay in email traffic is proportional to the sensing duration but with a smaller slope, which means that email traffic is less affected by the sensing duration than voice traffic. Moreover, email applications are not sensitive to delay. Video traffic is least sensitive to sensing duration, and the average delay and the sensing duration are not in a constant proportionality relationship. The results in Figure 5.21 demonstrate that the same sensing strategy has different effects on traffic from different applications.

The diverse traffic used in these scenarios follow different frame aggregation settings (see Section 4.3.3). Figure 5.22 shows the number of MPDU packets aggregated in one PPDU before being sent to the physical layer for all three types of traffic used in the simulations. For the video traffic, the PPDU contained two MPDUs, while for the voice traffic it contained only one. Sensing was conducted less frequently in the video traffic than the voice traffic, and therefore the video traffic was less affected by sensing duration. The email traffic used different PPDU sizes during simulation that on average was less than the size used for video traffic and slightly larger than the size used for voice traffic. It can also be seen in Figure 5.21 that sensing durations of less than 50 ms caused an average delay of fewer than 0.2 seconds, which is acceptable in most applications. However, a longer sensing duration could be required to achieve higher accuracy.

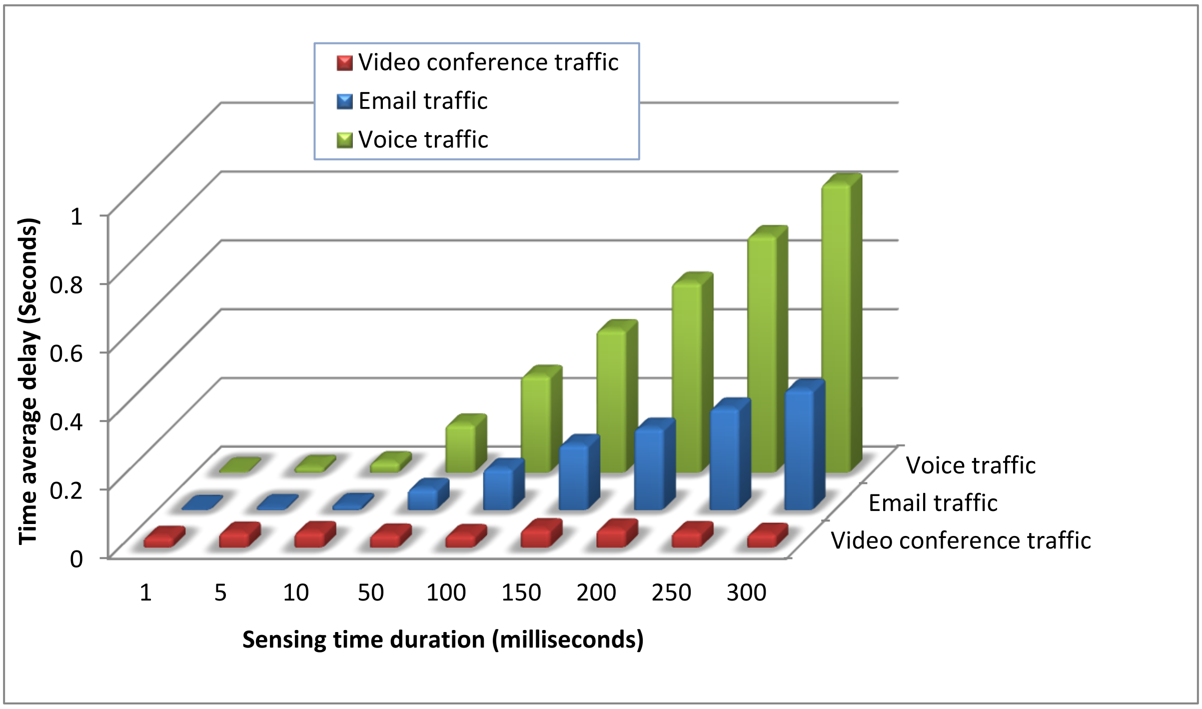


Figure 5.21 Average delay comparison between different applications' traffic and different sensing durations under fixed sensing strategy (infrastructure network with video conferencing traffic)

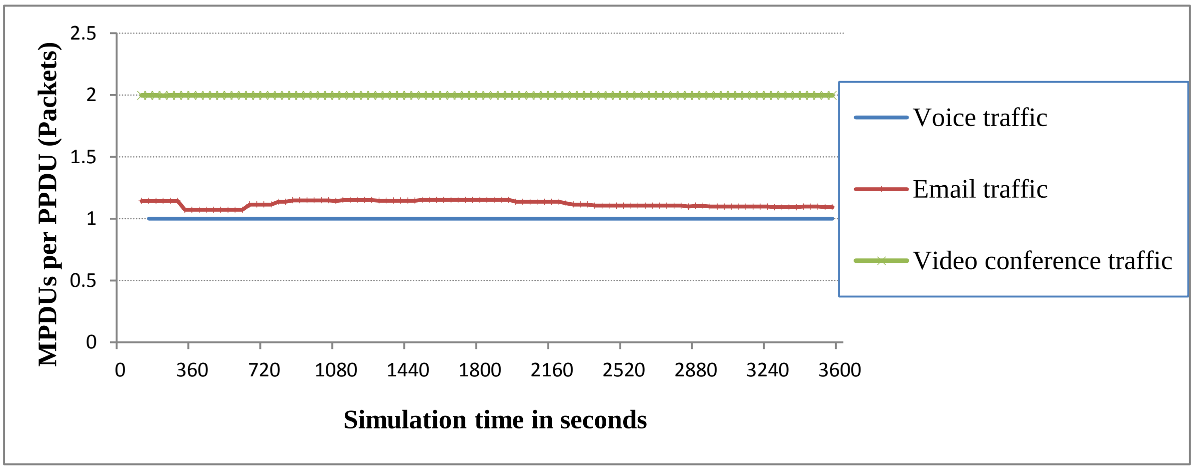


Figure 5.22 Comparison between the MPDU average size of the voice, email and video conferencing traffic under fixed sensing strategy (infrastructure network)

The ACK timeout is another factor that limits the available time for sensing before sending ACK when conducting sensing before all frames. The ACK timeout is calculated in the standard implementation as extended inter-frame space (EIFS) – DCF inter-frame space (DIFS), where EIFS = transmission time of ACK frame at lowest physical mandatory rate + SIFS + DIFS. Therefore, if the ACK timeout is shorter than the time required for the data frame to propagate to the receiver and the time taken by the ACK start to send back to the sender plus SIFS, then the sending MAC will assume that the packet has been lost and the sender will unnecessarily retransmit the data frame. Moreover, the retransmitted data frame may end up colliding with the late ACK frame, reducing throughput. In CR, the sensing duration has to be considered if the first sensing strategy is to be used, so the ACK timeout should be increased based on the sensing duration. Moreover, the CR may use frequency bands where the signal can propagate longer, and this increase in propagation delay should also be also included in the timeout. The same issues are encountered with CTS timeouts when the RTS/CTS mechanism is used to reduce collisions with hidden nodes.

5.4. Impact of imperfect sensing on QoS

Sensing accuracy is an important factor when comparing different sensing methods. One implication of sensing accuracy is the time required to achieve a high level of accuracy. A longer sensing duration allows for taking more samples from the sensed channel, which is an important parameter for achieving higher accuracy. Different sensing methods can achieve significantly diverse accuracy under the same spectrum environment, particularly in low SNR spectrum channels (see Section 2.3 for more detail). In this section, the impact of the main two measurements of sensing accuracy, P_d and P_f, on different applications is studied. The P_d represents the level of protection that the sensing method can achieve to avoid interference with PU transmission. The probability of missed detection (P_m); i.e., ($P_m = 1-P_d$), represents the probability of interfering with PU transmission because the sensing method fails to detect the PU presence. In contrast, P_f presents the missing probability of identifying the available spectrum holes in the sensed channels. A sensing method that gives a false alarm will force the CR device to vacate the channel and search for another.

To study the impact of P_d and P_f, the MAC process model 'wlan_dispatch_cr_noIFSP' and its child process model 'wlan_mac_hcf_cr_noIFSP' are implemented. In this new inquiry, the Sense state settings including the sensing duration and the P_d and P_f of the modelled sensing method. The sensing accuracy outcome of the Sense state, therefore, is now based on the P_d and P_f values. The network is designed to have four wireless workstation nodes and one wireless server node, with large distances between them, so channels experience low SNR (i.e., less than 15 dB; see Figure 5.23).

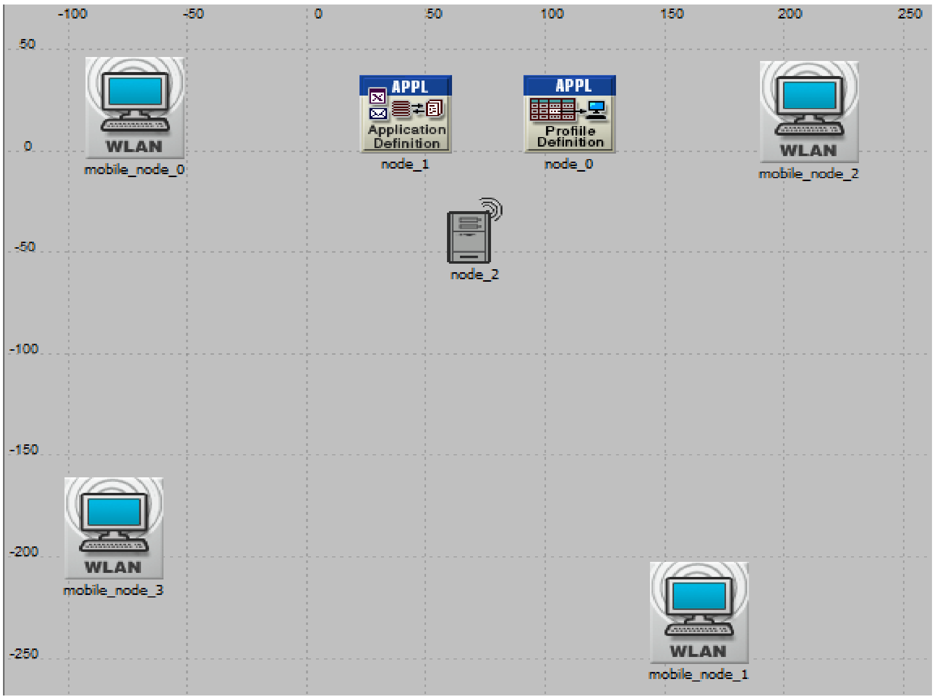


Figure 5.23 Network layout for studying imperfect sensing (scale in meters)

The received SNR in dB for a packet calculated in the Modeler is based on this equation:

/* Assign the SNR in dB. */

op_td_set_dbl (pkptr, OPC_TDA_RA_SNR, 10.0 * log10 (rcvd_power / (accum_noise + bkg_noise)));

Where rcvd_power is the received power after power attenuation because of propagation distance and (accum_noise + bkg_noise) is the accumulated noise levels calculated by the interference and background stages. Hence, the SNR can be decreased as a result of decreasing the received power when increasing the distance between nodes. Increasing interference or background noise also decreases the SNR. The workstation nodes were set to exchange IP telephony traffic with each other and heavy email traffic with the server. Node_2 acted as an AP and email server for the four mobile nodes. Node_1 was used to configure the applications, and node_0 was used to configure their profiles. The attributes of voice traffic are listed in Table 5.3. The attributes of the email application are listed in Table 5.4. The email application starts one session for generating emails and another for requesting emails. The two sessions operate for the entire simulation. The simulation parameters are listed in Table 5.5. The purpose of this section is to investigate the consequences of imperfect sensing without considering the PU's or other Secondary User (SU) systems' activities. The implication of different P_d and P_f on throughput and delay is studied in the following subsections.

Table 5.3 Attributes of IP telephony traffic

Attribute	Value
Silence Length (seconds): **defines the time in seconds spent in speech mode by the called party.**	Exponential (0.65)
Talk Spurt Length (seconds): **defines the time in seconds spent in speech mode by the calling party.**	Exponential (0.65)
Encoder Scheme	G.729 A
Voice Frames per Packet	1
Compression Delay (seconds)	0.02
Decompression Delay (seconds)	0.02
Voice Conversation Environments	All environments

Table 5.4 Attributes of email application traffic

Attribute	Value
Send Inter-arrival Time (seconds): **defines when the next group of emails is sent. The start time of the next group of emails is computed by adding the inter-arrival time to the time at which the previous email group was completed.**	Exponential (360)
Send Group Size: **defines the number of 'queued emails' to be sent.**	Constant (3)
Receive Inter-arrival Time (seconds): **defines the amount of time between receiving emails. The start time for the next email reception is computed by adding the inter-arrival time to the time at which the previous emails were received.**	Exponential (360)
Receive Group Size: **defines the number of 'queued emails' to be received.**	Constant (3)
E-Mail Size (bytes)	Constant (2000)

Table 5.5 Simulation parameters

Parameter	Value
Capture mode:	Bucket
Sample mean:	Total of 100 values.
Seed:	128
Updated interval:	500000 events

5.4.1. Imperfect sensing and throughput

In this subsection, the impact of imperfect sensing on the achieved throughput is investigated. The sensing duration was fixed to 5 ms for all scenarios, but the P_f and P_d were changed in each one. A moderate sensing duration of 5 ms was chosen to reduce this factor's effect, as it had been studied in previous sections. The calculated average throughputs achieved by Node_0 (Figure 5.23) under a sample of different sensing accuracies ($P_f = 1$, P_d = 0.5), ($P_f = 1$, P_d = 0.4), ($P_f = 0.95$, P_d = 0.5) and ($P_f = 0.95$, P_d = 0.4), are shown in Figure 5.24. The results reflect the impact of sensing on the throughput QoS metric where P_d and P_f are slightly diverse, by 0.05 and 0.1 respectively. For a dramatic change in sensing accuracy, Figure 5.25 shows the noticeable difference in average throughput when high- and low-accuracy sensing are used.

When comparing high-accuracy sensing, as shown in Figure 5.24, for the same P_d the node can achieve higher average throughput when P_f is decreased. In contrast, decreasing the P_d has less impact on throughput than P_f; the PU is not present in these scenarios. Under the same P_f value, using sensing with less P_d may even lead to slightly better throughput when the PU is absent. For instance, when P_f = 0.4 sensing at P_d = 0.95 starts giving higher throughput than sensing with P_d = 1 after ten minutes of simulated time (see Figure 5.24). In Figure 5.25, the low accuracy sensing with low P_d still can achieve acceptable throughput as long as P_f is not very high. However, when P_f reaches a high value, for instance, 0.9, the result is extremely low throughput. Figure 5.26 shows that a low value of P_f can help in achieving higher throughput as P_f = 0.001 can achieve nearly double the throughput of P_f = 0.1 under the same P_d = 0.5. For the same sensing method, accuracy varies according to the SNR and, thus, the achieved throughput. The SNR is one of the fundamental factors that affect the achieved throughput even for the same accuracy. For instance, Node_0 and Node_1 achieved different throughput under the same sensing duration (5 ms) and accuracy (P_d = 0.95 and P_f = 0.4), as shown in Figure 5.27. Node_0 achieved higher throughput than Node_1 as its SNR is higher (see Figure 5.28).

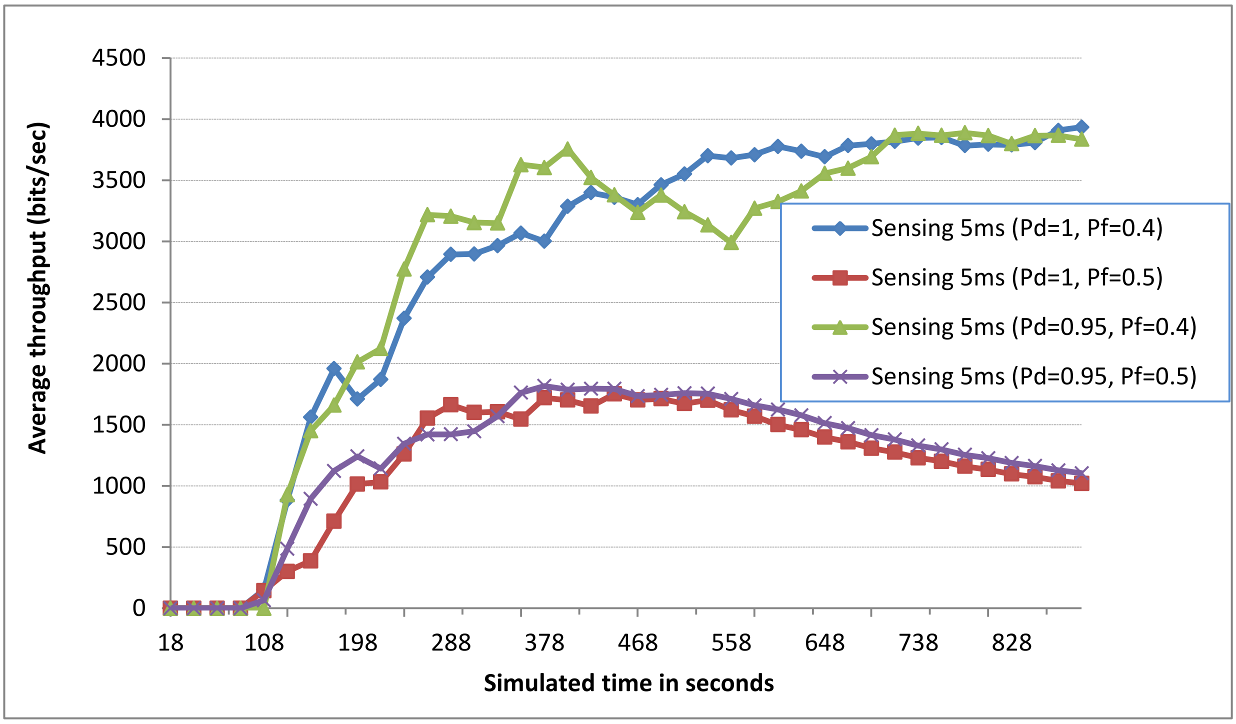


Figure 5.24 The average throughput of different high sensing accuracies (mobile_ node_0)

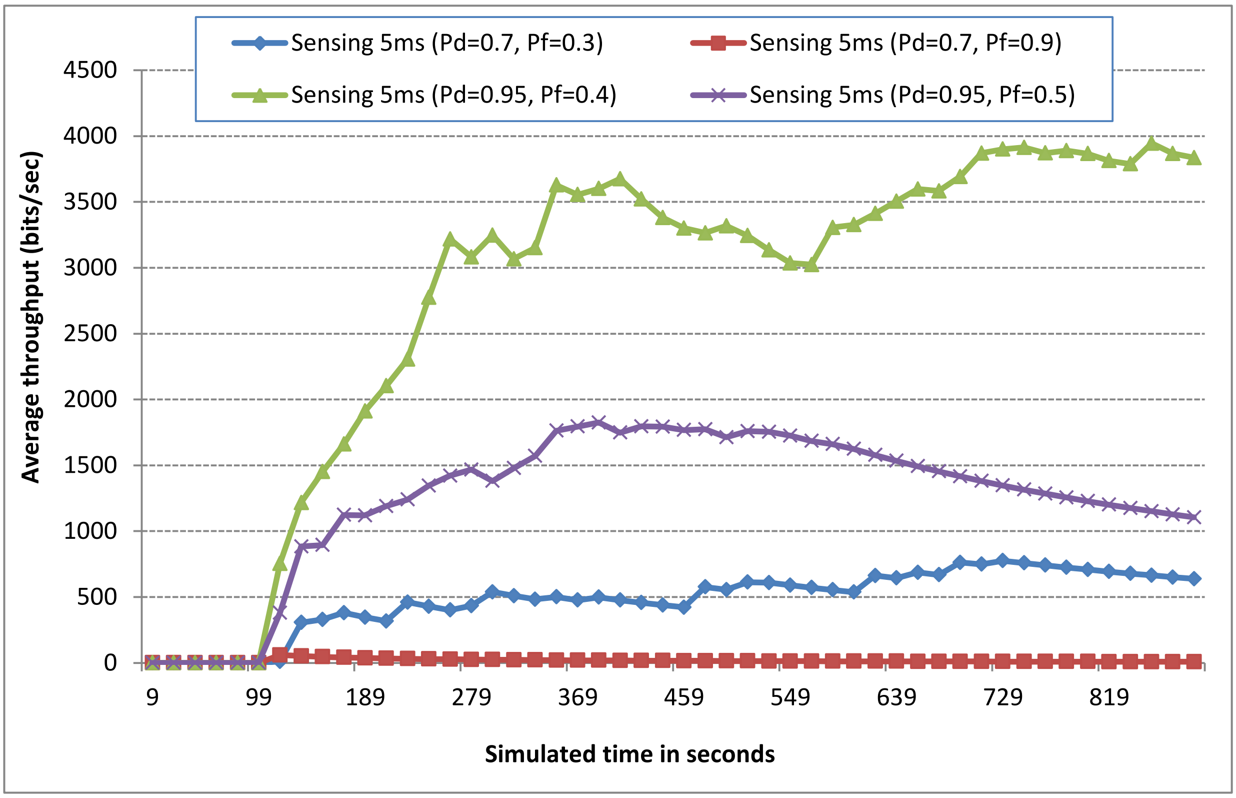


Figure *5.25* The average throughput of different high and low sensing accuracies (mobile_ node _0)

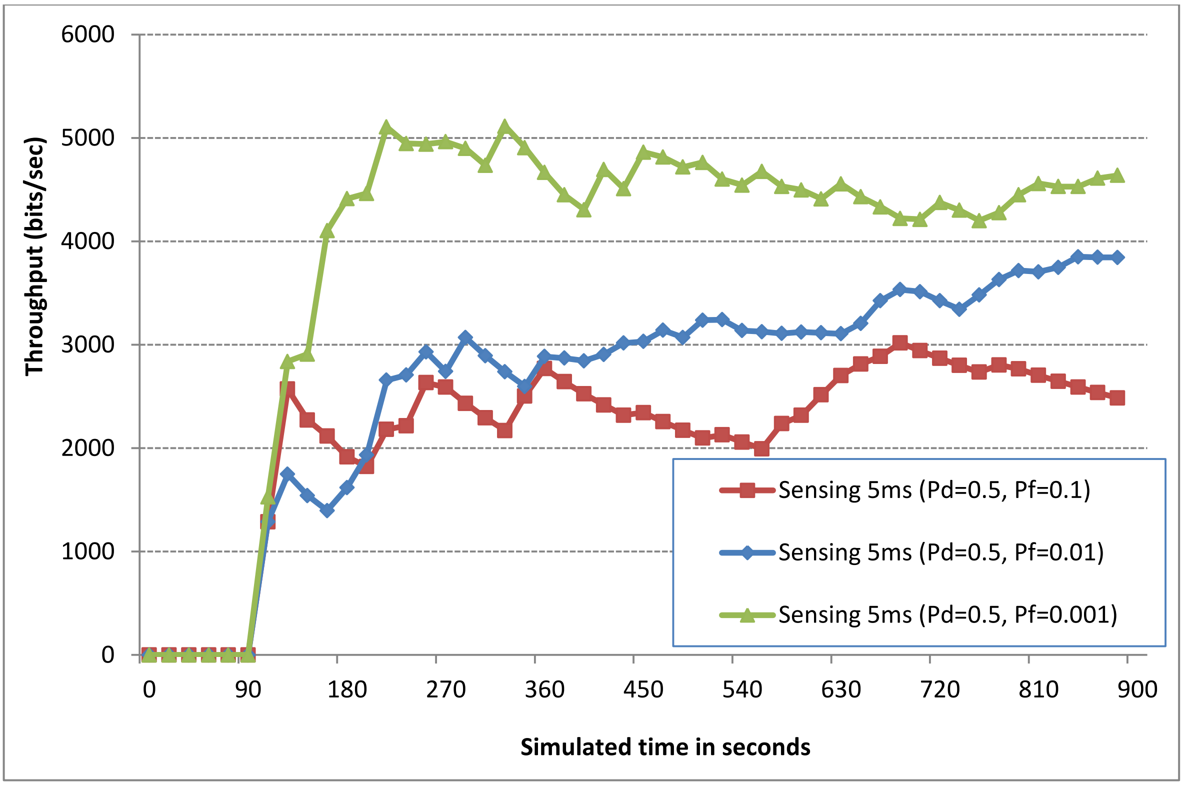


Figure 5.26 Average throughput under different P_f values for P_d = 0.5 (mobile_ node _1)

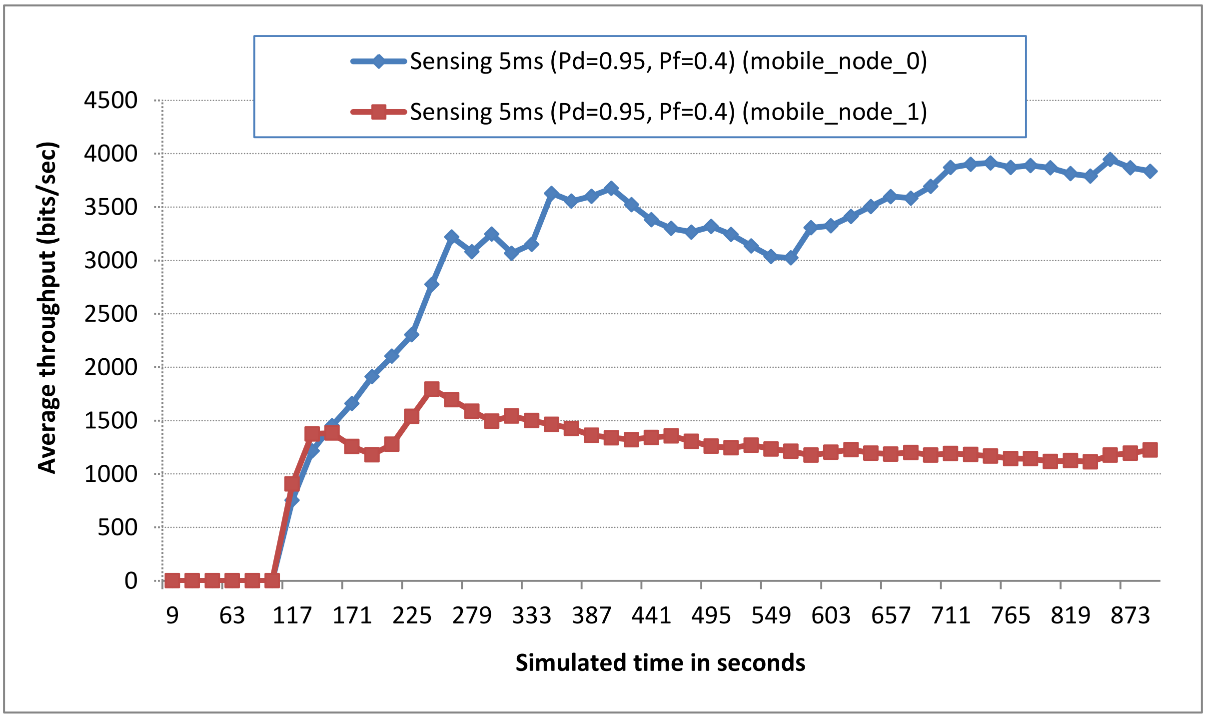


Figure 5.27 Comparing the average throughput of mobile_ node _0 and mobile_ node _1

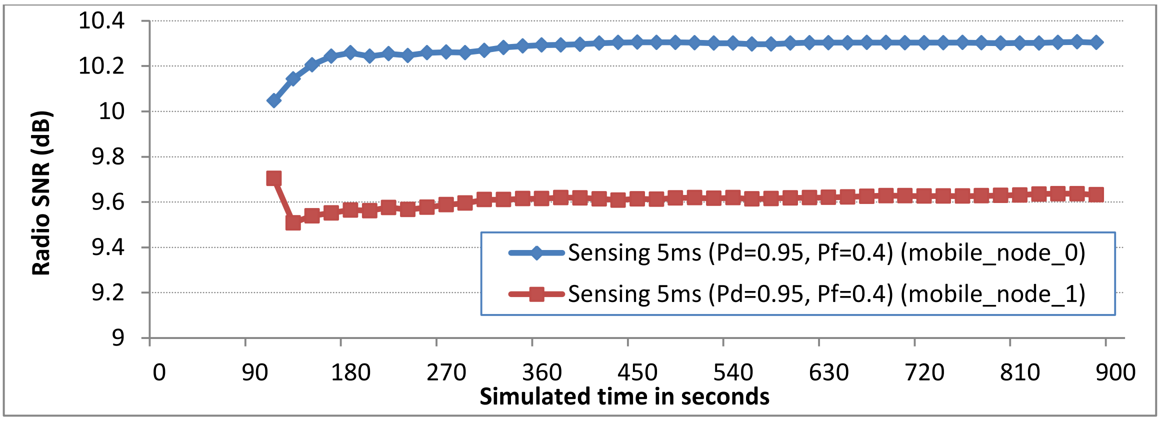


Figure 5.28 Comparing SNR (dB) of two mobile nodes

5.4.2. Imperfect sensing and delay

Sensing is an important QoS metric for real-time applications. In this subsection, different types of delay statistics that can be measured by Riverbed Modeler were analysed under different imperfect sensing values.

5.4.2.1. Media access delay

The media access delay reflects the time overhead of transmitting a packet at the MAC layer process. In Section 5.3, the results show a significant impact of sensing duration when perfect accuracy is assumed. In this section, a sample of the measured media access delay of different sensing accuracies is illustrated in Figure 5.29.

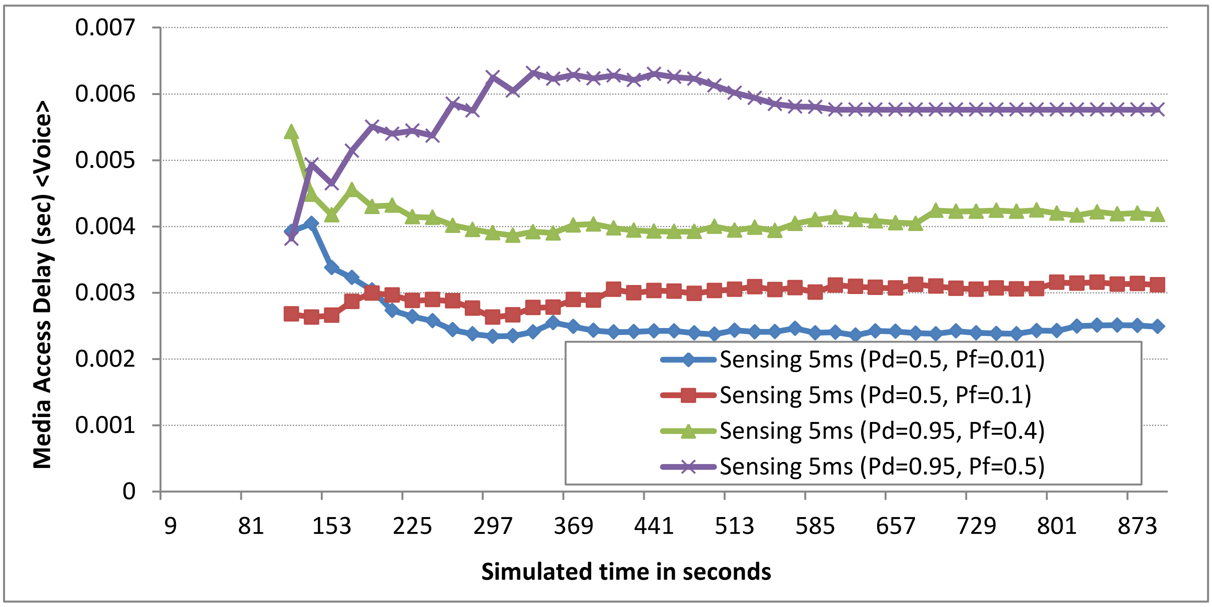


Figure 5.29 Media access delay when S_d = 5 ms with high and low accuracies (mobile_ node _3)

The media access delay is slightly affected by P_d and P_f changes for the same sensing durations. For the P_d = 0.95, the media access delay increased about 0.0015 seconds to reach 0.0057 seconds when P_f was increased from 0.4 to 0.5. For P_d = 0.5, the media access delay increased by about 0.0006 seconds when P_f was increased from 0.01 to 0.1.

5.4.2.2. Voice packet end-to-end delay

The Riverbed Modeler provides the option to gather the statistics related to the total voice packet delay, called 'analog-to-analog' or 'mouth-to-ear' delay, per an advanced WLAN node. This total delay comes from the sum of network delay, encoding delay, decoding delay, compression delay, decompression delay and de-jitter buffer delay. Network delay is from the time the sender node sends the packet to the real-time transport protocol (RTP) to the time the receiver gets it from RTP. Encoding delay is computed on the sender node based on the used encoder scheme (see Table 5.3). Decoding delay is calculated on the receiver node and is assumed to be equal to the encoding delay. Compression and decompression delays are set in the corresponding attributes in the Voice application configuration (see Table 5.3). The de-jitter delay is determined based on the Voice conversation environments settings under the mean opinion score (MOS) in the Voice application configurations (see Table 5.3).

A sample of the resulting packet end-to-end delay of voice traffic from mobile_node_2 to mobile_node_3 when the sensing accuracies are P_d = 0.5 and P_f = 0.01; P_d = 0. 5 and P_f = 0.001; P_d = 0.95 and P_f = 0.4; and, P_d = 0.95 and P_f = 0.5, are compared in Figure 5.30. The results show no significant change in end-to-end delay when the sensing accuracy is changed under the same sensing duration for the same wireless less link. In contrast, a noticeable difference between end-to-end delay is measured in different wireless links under the same sensing duration, 5 ms, and accuracy, P_d = 0.5 and P_f = 0.1, as shown in Figure 5.31. The end-to-end delay differences are mainly because of the different distances between the network nodes. For instance, the distance between mobile_node_0 and mobile_node_1 is longer than that between mobile_node_2 and mobile_node_3. The measured delays were very high and not acceptable for voice calls; that is because the distances between nodes were large to have low SNR links between nodes.

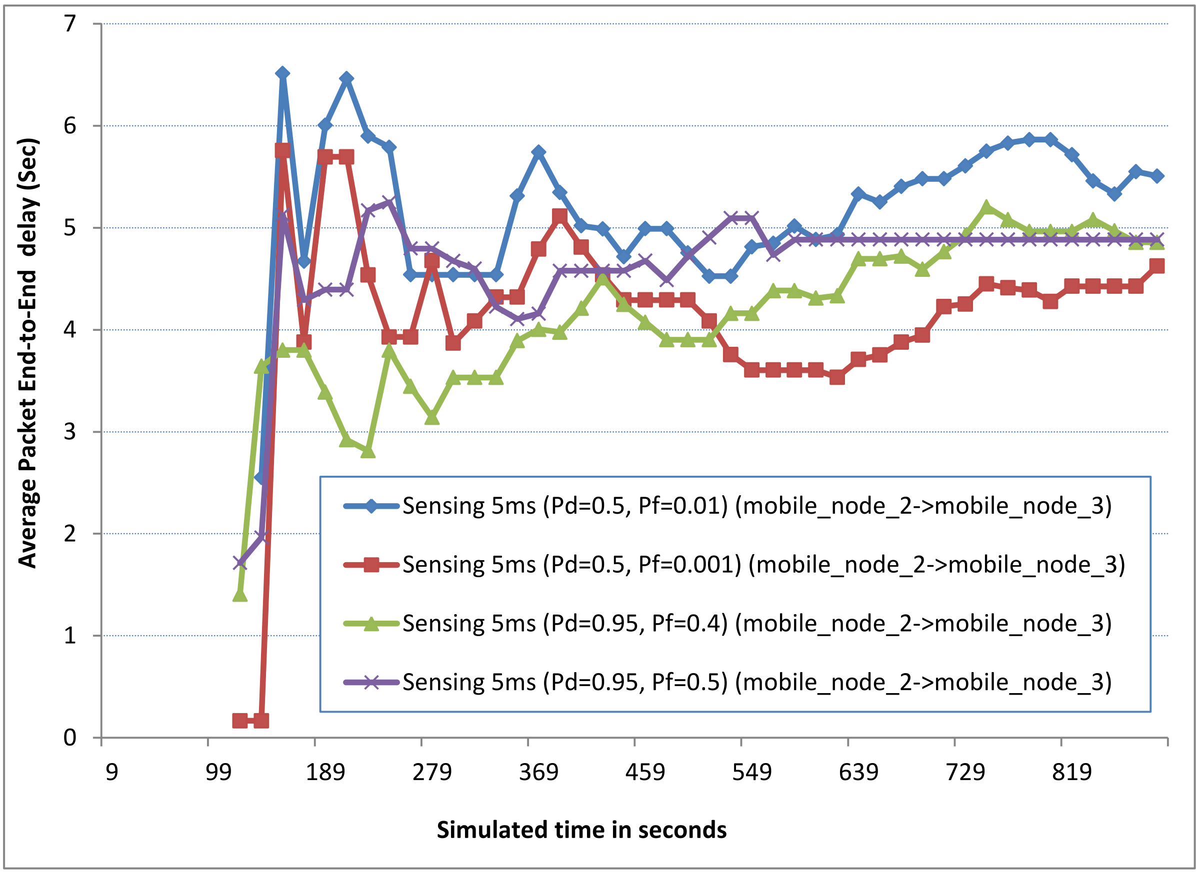


Figure 5.30 Average packet end-to-end delay of the voice traffic under different sensing accuracies (S_d = 5ms)

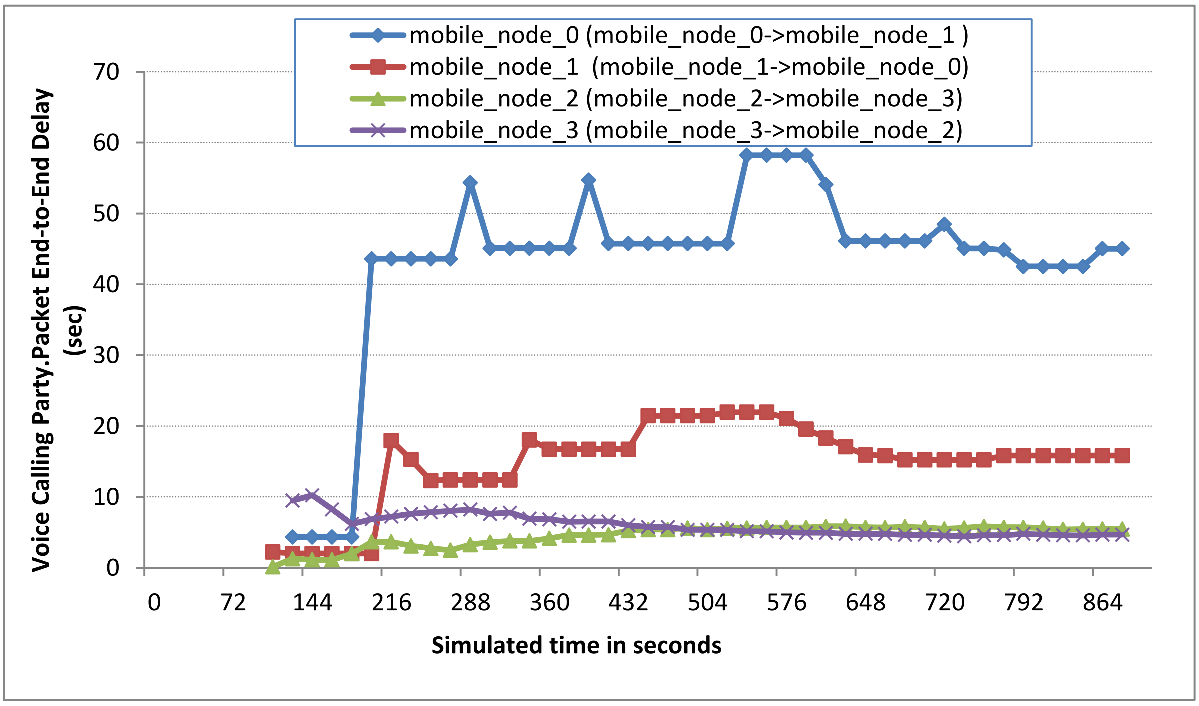


Figure 5.31 Average voice packet end-to-end delay when S_d = 5ms (P_d=0.5, P_f=0.1)

5.4.2.3. Voice packet delay variance

Voice packet delay variance is also can be calculated and collected during the Riverbed simulation for voice application when an advanced node is used. Packet delay variance in a node is the variance among end-to-end delay for voice packets received by this node. For this measurement, it is measured from the time it is created to the time it is received. The delay variance is computed by the simulation tools based on the below equation [209]:

Variance = [Sum (end-to-end_delay_Sample^2) / number_of_samples] – [Sum (end-to-end _delay_Sample) / number_of_samples]^2

Samples of the results of the calculated average voice packet delay variance under different sensing accuracies are presented in Figure 5.32 and Figure 5.33. The results show that the higher the sensing accuracy the lower the delay variance, under the same conditions. In Figure 5.32 the delay variance calculated for the received voice traffic by mobile_node_3 is lowest when P_d = 1 and P_f = 0.4 for the 5 ms sensing, compared to the other less accurate sets: P_d = 0.1 and P_f = 0.5; P_d = 0.95 and P_f = 0.4; and P_d = 0.95 and P_f = 0.5. The delay variance increases as the sensing accuracy is decreased, to reach its nadir when P_d = 0.95 and P_f = 0.5. The same was found when P_f was 0.1 and 0.01, when P_d is low at 0.5: the lower the P_f the less delay variation, as shown for the voice traffic between mobile_node_2 and mobile_node_3 in Figure 5.33.

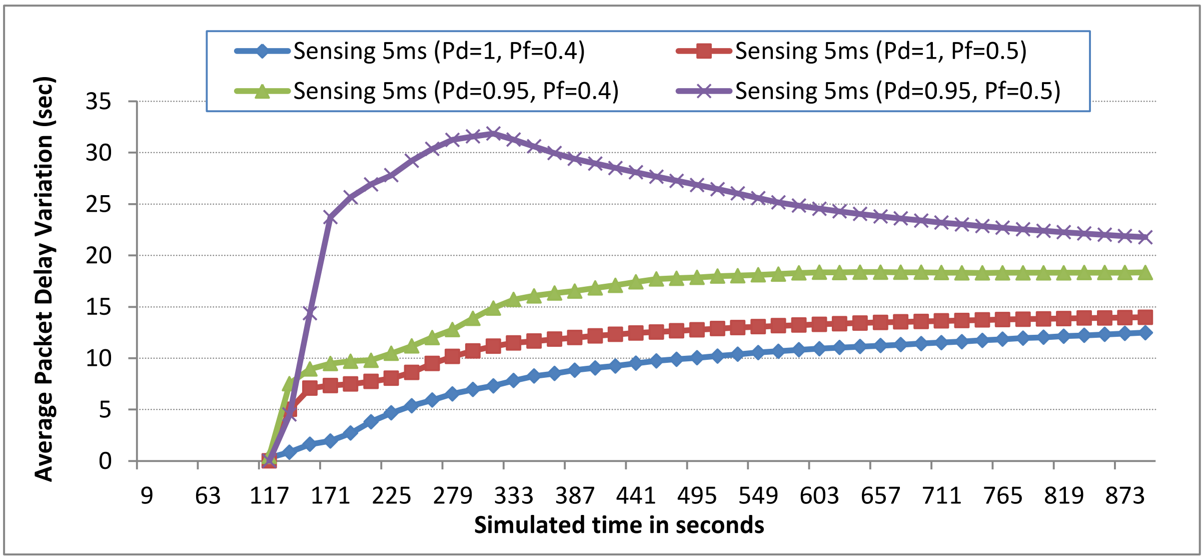


Figure 5.32 Average voice packet delay variation under different high P_f when S_d = 5ms (mobile_node_3)

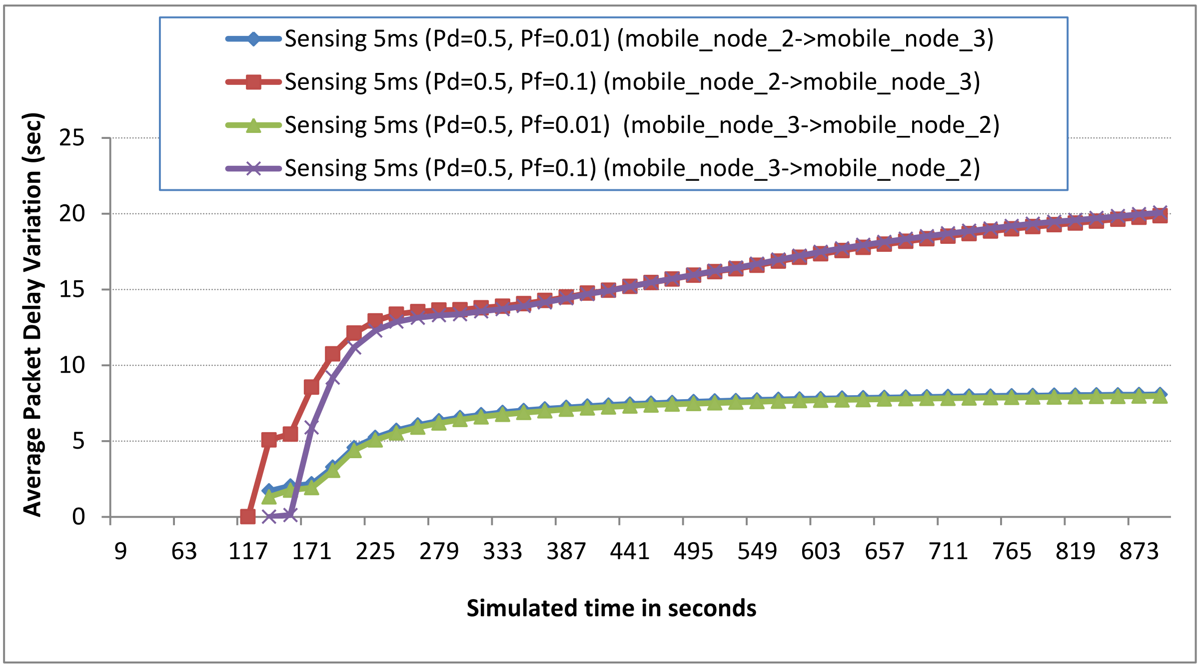


Figure 5.33 Average voice packet delay variation under low P_f

5.4.2.4. Jitter

For voice applications, jitter is an important QoS measurement, and the Jitter statistics were set to be gathered and calculated by the simulation tool for the voice application. If two consecutive voice packets leave the source node with timestamps t_1 and t_2 and are played back at the destination node at times t_3 and t_4, the jitter is $(t_4 - t_3) - (t_2 - t_1)$. Positive jitter indicates that the second packet experienced a higher delay in reaching its destination than the first one. Negative jitter indicates that the time difference between the packets at the destination node was less than that at the source node.

The results show that jitter is sensitive to sensing accuracy. The average jitters observed for voice traffic from mobile_node_2 to mobile_node_3 are represented in Figure 5.34 and Figure 5.35 as examples. The results show that, for the same P_d, a higher P_f returns higher jitter. For instance, sensing with P_d = 0.95 and P_f = 0.5 caused higher jitter along the simulation than when P_d = 0.95 and P_f = 0.4, as shown in Figure 5.34. The same finding is made when we compare the jitter measured when P_f = 0.01 with P_f = 0.001 for the same P_d = 0.5, as shown in Figure 5.35. The jitter for P_f = 0.001 was the lowest of the illustrated results. Increasing sensing accuracy will help decrease jitter under the same operating conditions.

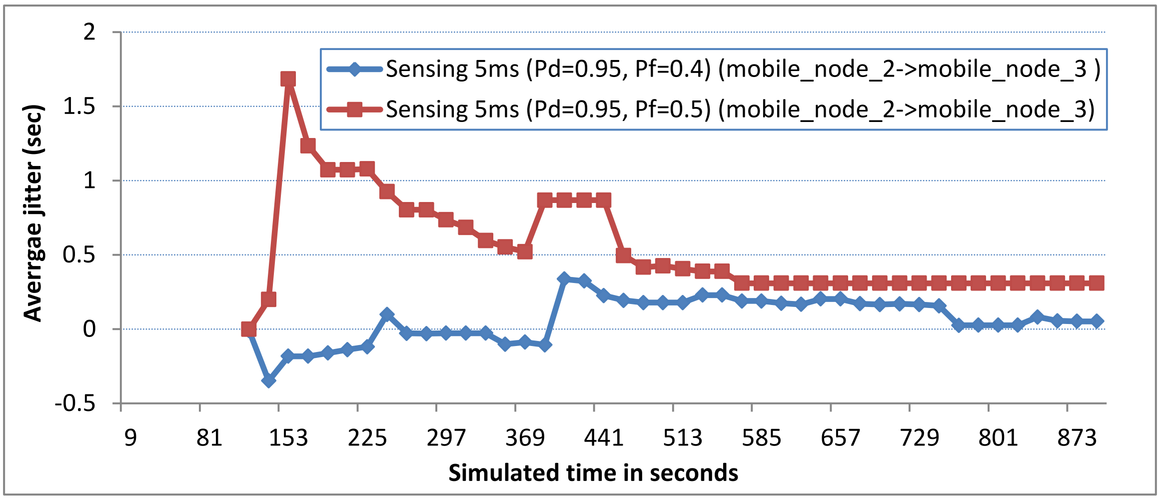


Figure 5.34 Average jitter for voice traffic when S_d = 5ms (P_d = 0.95, P_f = 0.4 and 0.5)

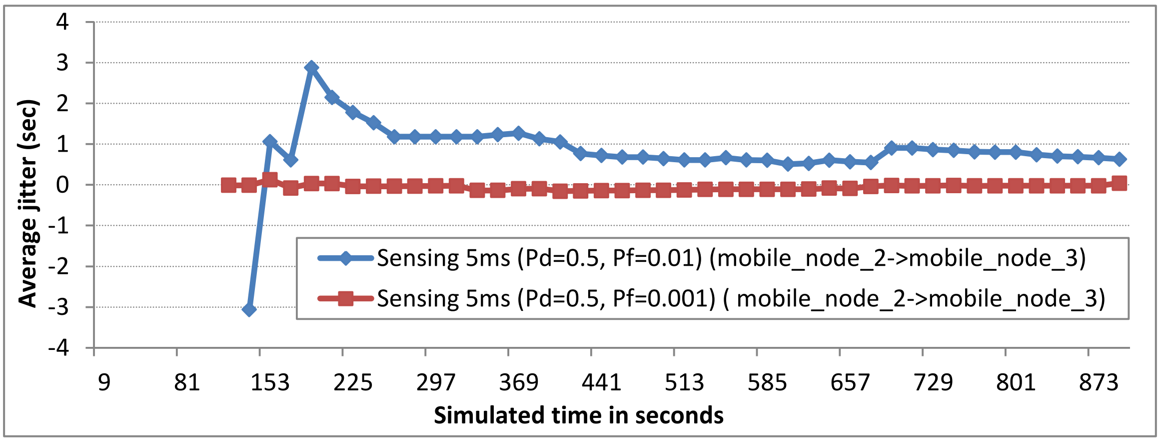


Figure 5.35 Average jitter for voice traffic when S_d = 5ms (P_d = 0.5, P_f = 0.001 and 0.01)

5.5. Evaluating the proposed sensing strategies

In this section, several simulation scenarios are used to evaluate the proposed sensing strategy solutions. First is the general proposed fuzzy logic sensing selection mechanism explained in Chapter 3; for reference, it is called ‘select sensing strategy’. Second is the QACR-MAC strategy proposed in Chapter 4 as a customised version of the general solution for White-Fi networks, to enhance QoS and the IEEE 802.11e mechanism. The focus is on evaluating the QACR-MAC strategy. In the subsection below the proposed solutions are evaluated to show how QACR-MAC noticeably improves the overall QoS in a CR network based on IEEE 802.11. In the following subsections, QACR-MAC is evaluated for IEEE 802.11e QoS enhancement, different frame aggregation mechanisms, coexistence, and varying numbers of CR nodes.

5.5.1. *Evaluation of the proposed sensing strategies*

In this section, several simulations conducted to evaluate the QoS enhancement that can be achieved by considering the application's requirements for an appropriate sensing strategy. The simulation scenarios were implemented to compare the effects of fixed sensing, selecting the sensing, and QACR-MAC approaches. The network in all scenarios was the same, with four nodes and one server, as shown in Figure 5.17. The application definition (Node_1) was used to define simultaneously three applications: IP telephony voice, high-resolution video conferencing, and heavy load email. The profiles of the defined applications were configured in the profile definition (Node_0) as shown in Figure 5.36. The three applications were configured to run simultaneously on all nodes; a snapshot of configuring mobile_nodes is shown in Figure 5.37. The WLAN parameters were similar to those shown in Table 5.2.

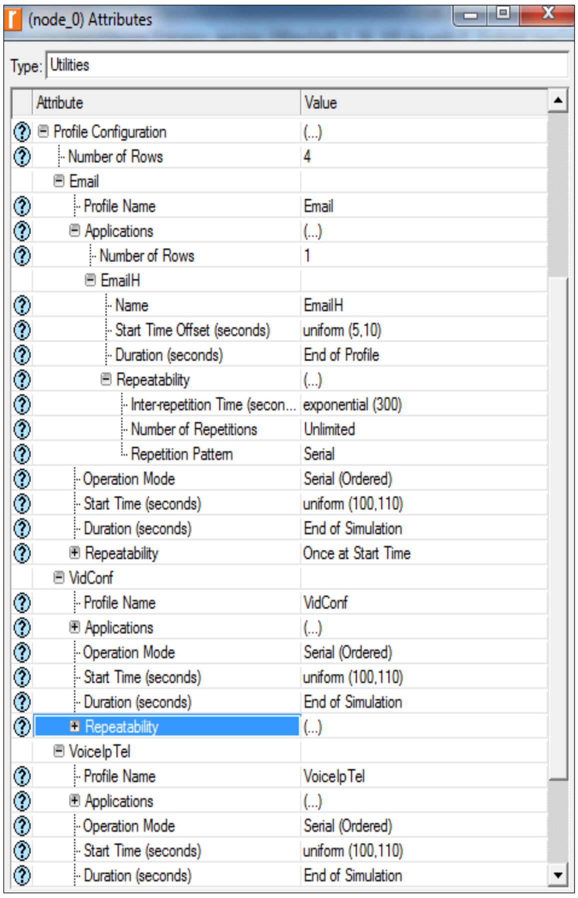


Figure 5.36 Profile configurations for email, video conferencing and voice IP telephony applications

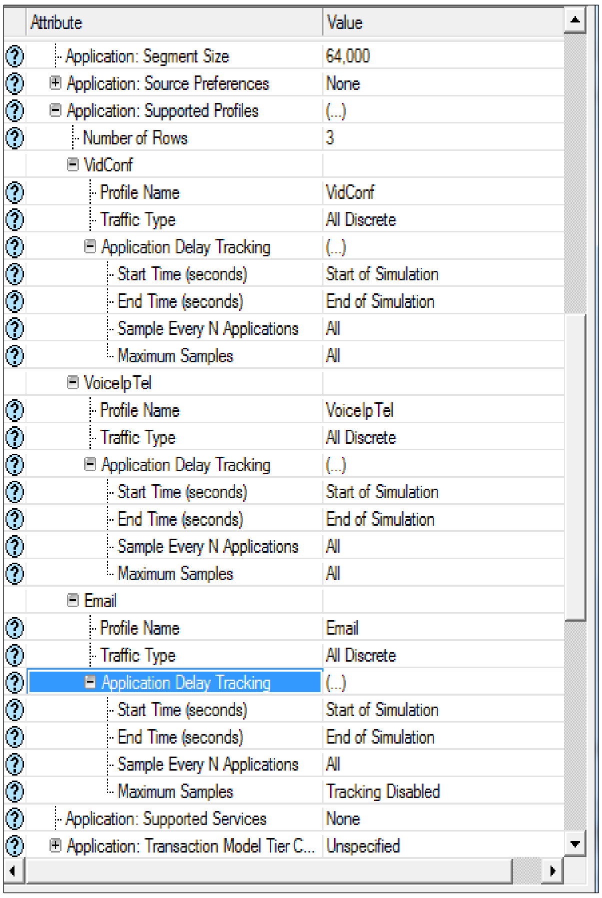


Figure 5.37 Configuring nodes with the defined application profiles

For the fixed sensing strategy, the nodes were configured to use the customised MAC process model 'wlan_dispatch_cr_noIFS' and the fixed duration sensing was conducted for all frames except response frames, which can be considered the conventional approach for sensing strategies in 802.11af networks. Under this approach, the simulations were run for different fixed sensing durations. In each run duration was set to 1, 5, 10, 15, 20, 25, 30, 35, 40, 45, 50, 60, 100, 150, 200, 250, 300, 350, 400, 450 or 500 milliseconds. Sensing accuracy was assumed perfect in all simulations.

To evaluate the sensing selection approach proposed in Chapter 3, the nodes were configured to use the customised MAC process model 'wlan_dispatch_cr_noIFSPS' and the nodes were implemented to select $S_d[AC]$ based on the AC of the frame, such as S_d[AC_VO] = 1 ms, S_d[AC_VI] = 5 ms, S_d[AC_BE] = 50 ms and S_d[AC_BK] = 100 ms, as shown in Table 5.6. The selection was applied before sending all frames except response frames. The sensing accuracy was assumed to be perfect in all durations, and other factors affecting the sensing selection were fixed (CR capability is high, prior information is high and PU protection is low).

Table 5.6 The sensing duration for each AC

Frame Access Categories	Sensing mode	Sensing duration ($S_d[AC]$) in milliseconds (ms)
Voice (AC_VO)	Coarse sensing	S_d[AC_VO] ≤ $\mathbf{Thr_1}$; i.e., S_d[AC_VO] = 1 ms
Video (AC_VI)	Moderate sensing	$\mathbf{Thr_1}$<S_d[AC_VI] ≤ $\mathbf{Thr_2}$; i.e., S_d[AC_VI] = 5 ms
Best effort (AC_BE)	Fine sensing	$\mathbf{Thr_2}$ < S_d[AC_BE] ≤ $\mathbf{Thr_3}$; i.e., S_d[AC_BE] = 50 ms
Background (AC_BK)	Extra Fine sensing	$\mathbf{Thr_3}$< S_d[AC_BK] ≤ $\mathbf{S_{dMAX}}$; i.e., S_d[AC_BK] = 100 ms

For evaluating the QACR-MAC proposed in Chapter 4, the nodes were configured to use the customised MAC process model 'wlan_dispatch_cr_noIFSPS4'. As in the sensing selection approach, the nodes were implemented to select $S_d[AC]$ based on the AC of the frame, such as S_d[AC_VO] = 1 ms, S_d[AC_VI] = 5 ms, S_d[AC_BE] = 50 ms and S_d[AC_BK] = 100 ms, as shown in Table 5.6. However, the selection mechanism was only employed for the first attempt at transmitting a frame, and otherwise normal ED sensing was conducted. As with the previous approaches, the sensing was conducted before transmitting response frames. The simulations were conducted under the same assumptions and settings as the previous approach.

A sample of the overall achieved average throughput by mobile_node_1 is shown in Figure 5.38, for fixed sensing when S_d was 1 ms and 100 ms, and for select sensing and QACR-MAC strategies. The proposed QACR-MAC reached noticeable improvement in throughput, but the proposed selection sensing strategy achieved a throughput slightly higher than the fixed approach when S_d = 1 ms. In the fixed sensing approach, increasing the sensing duration leads to lower throughput: we can see a significant difference in the achieved average throughput between 1 ms and 100 ms in Figure 5.38.

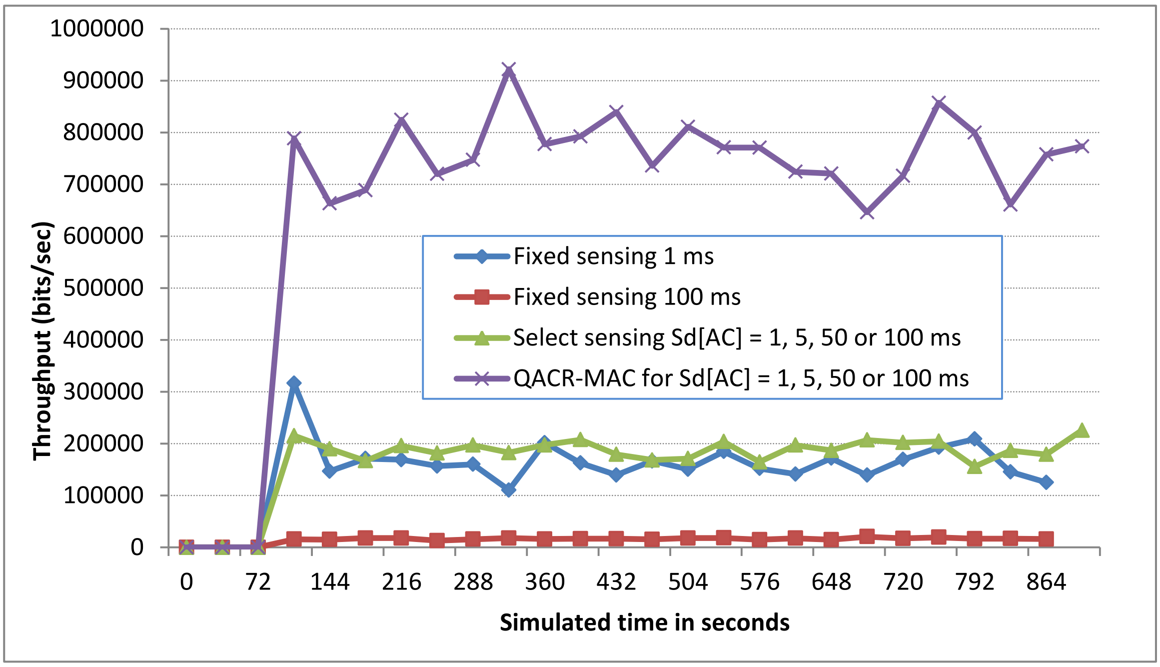


Figure 5.38 Throughput for different fixed sensing durations and our proposed sensing strategy (voice, video and email applications running concurrently mobile_node_1)

The delay was measured for all simulations. This delay presents the end-to-end delay of all data packets that are successfully received by the MAC layer and forwarded to the higher layer. In Figure 5.39 the measured delays for fixed sensing durations 1 ms and 100 ms are compared with those for the select sensing and QACR-MAC strategies in mobile_node_1. As demonstrated in Section 5.3, under the fixed sensing strategy the delay will increase as the sensing duration increases. However, the measured delays when using proposed sensing strategies were noticeably less than the delay measured for fixed duration 1 ms, as shown in Figure 5.39. Among the running applications, the IP telephony voice was the most sensitive to delay, suggesting that more focus on the delay of the voice packets is required to compare select sensing strategy and its improvement QACR-MAC. An example of such a comparison is shown in Figure 5.40 for the measured voice packet delays in mobile_node_1, between select sensing strategy and QACR-MAC, showing that using QACR-MAC caused less delay, around 17.25 ms, than the select sensing strategy.

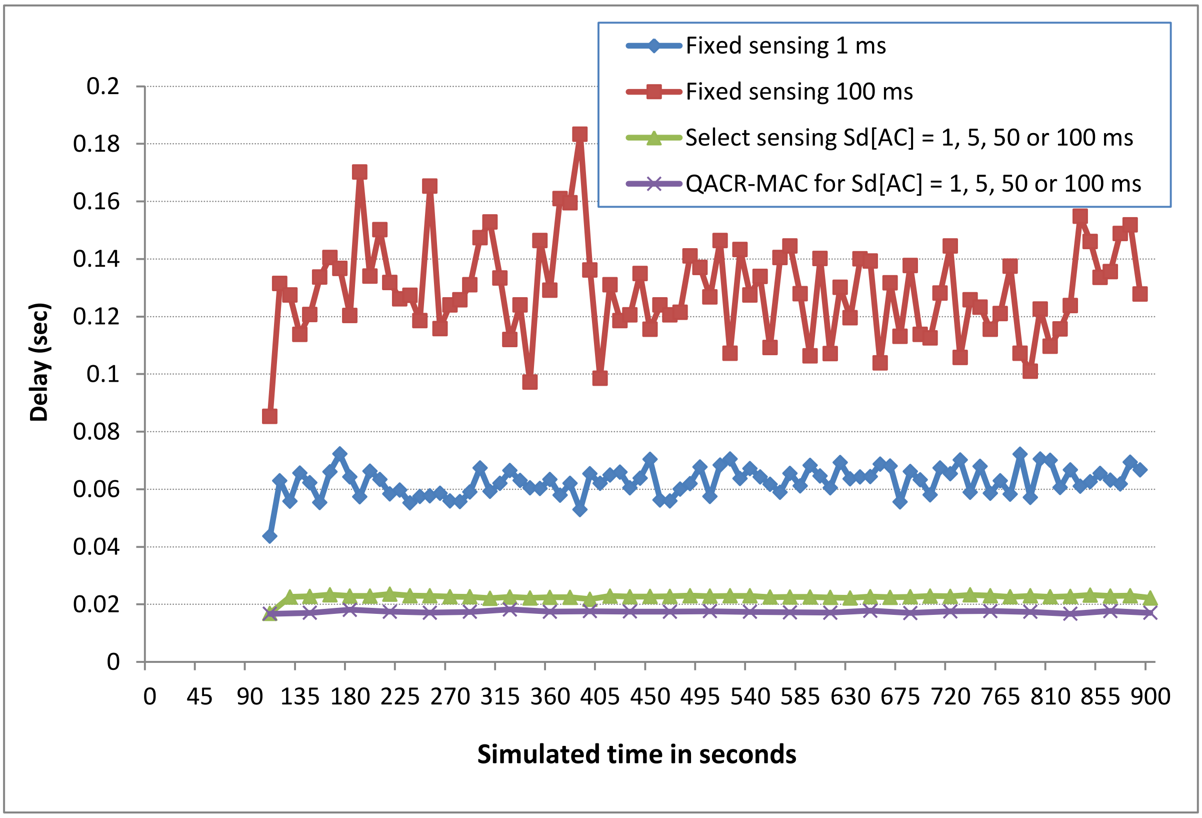


Figure 5.39 The measured delay under fixed sensing, select sensing and QACR-MAC strategies (voice, video and email applications running concurrently for mobile_node_1)

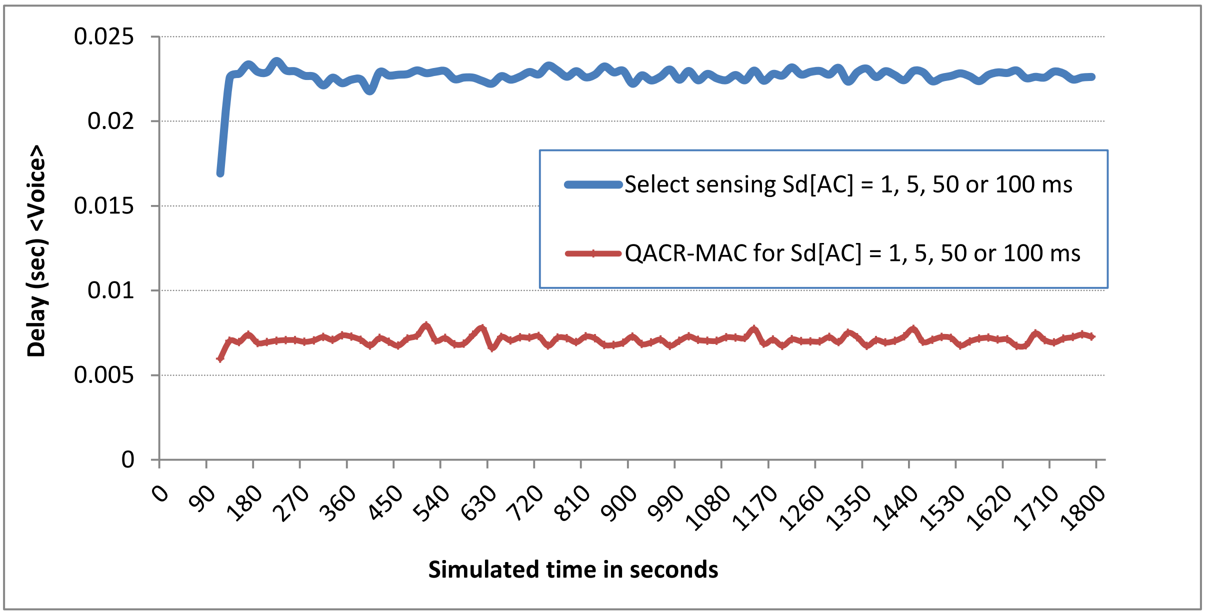


Figure 5.40 Comparison of the select sensing strategy and QACR-MAC in the measured voice packet delays (mobile_node_1)

The results of this section show that under the fixed approach a short sensing duration, e.g., 1 ms, resulted in less delay and higher throughput than a longer duration, e.g., 100 ms. The simulations were conducted assuming perfect sensing. In reality, the fixed small sensing duration most likely results in inefficient spectrum utilisation and less protection to PU signals particularly, for blind sensing methods and under low SNR for the wireless channels. The use of different sensing methods, durations and SNR leads to different degrees of imperfect sensing. In Section 5.4, the imperfect sensing impact on the delay and other QoS metrics are studied and analysed.

In this section, the simulation results demonstrate that the proposed select sensing strategy and its customisation QACR-MAC can achieve higher QoS than the fixed sensing strategy even when perfect sensing is assumed for small duration sensing. Although the select sensing strategy and QACR-MAC were both involved in higher sensing durations (e.g., 5 ms, 50 ms and 100 ms), the QoS degradation caused by using these higher durations was minimised. The results of this section prove that the proposed sensing strategies outperform the best know strategies, which eventually considered as fixed sensing strategies, even when perfect sensing is assumed.

5.5.2. Sensing strategy QACR-MAC and IEEE 802.11e

In Section 4.3.5, the IEEE 802.11e mechanism for enhancing the QoS in 802.11 wireless networks is briefly described. This differentiates between different traffic categories when accessing the wireless channel, to enhance the achieved QoS in IEEE 802.11, including White-Fi networks. However, when spectrum assessment relies solely on sensing in White-Fi networks, the impact of high-accuracy sensing on IEEE 802.11e has not been intensively studied. Under the fixed sensing approach, increasing the sensing duration to attain higher accuracy is expected to compromise the IEEE 802.11e mechanism. The results presented in Section 5.3.2 show that the efficiency of IEEE 802.11e degrades as the sensing duration is increased. For instance, voice traffic starts experiencing higher delays than other traffic as the sensing duration is increased (see Figure 5.21). Although IEEE 802.11e was enabled, its mechanism failed to achieve lower latency for voice traffic, compared to other types of traffic, in particular, for longer sensing durations. The IEEE 802.11e mechanism is expected to be more negatively impacted if the sensing duration is changed without considering QoS requirements.

In this section, another set of simulation scenarios is used to study the impact of using the QACR-MAC sensing strategy on IEEE 802.11e and compare that with the fixed sensing approach. The layout of the wireless network used for this study is shown in Figure 5.41. The network consists of eight wireless nodes and one AP, called QAP. The node settings are adopted from a scenario provided by the Riverbed Modeler, called e-Study, to study the default settings of IEEE 802.11e used to differentiate between different AC traffic for QoS enhancement. The eight wireless nodes are combined into four pairs, each configured with different AC traffic: background, best effort, video or voice. A snapshot of the configuration for each of these is presented in Figure 5.42. A snapshot of the common WLAN settings for all nodes is shown in Figure 5.43. All nodes were configured to use the default IEEE 802.11e settings, as shown in the section on EDCA parameters in Table 5.2. The simulated time was 720 seconds for all scenarios, under the simulation parameters listed in Table 5.5.

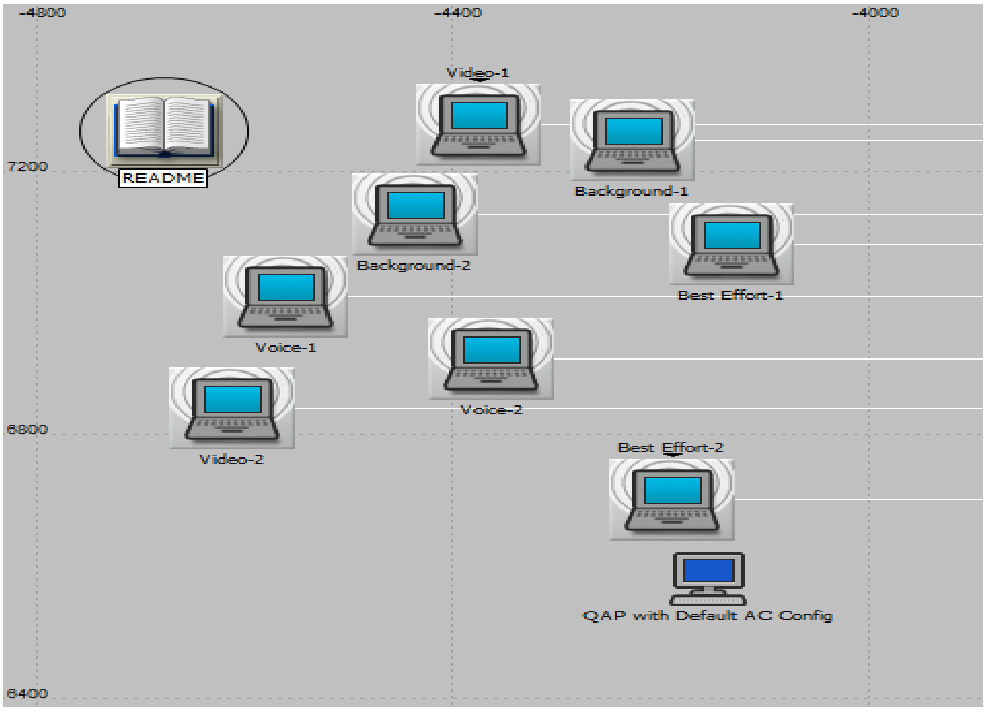


Figure 5.41 Network layout for studying QACR-MAC impacts on IEEE 802.11e (scale in meters)

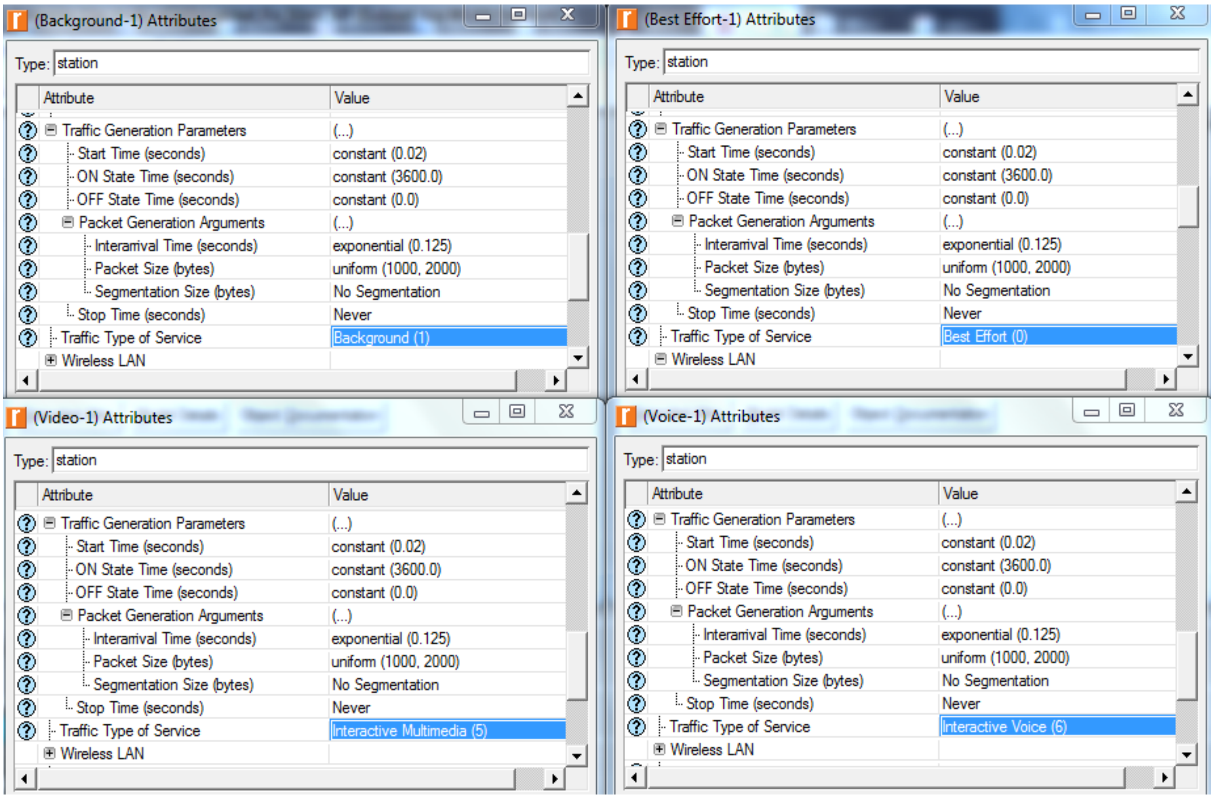


Figure 5.42 Traffic configuration for each type of AC traffic

Attribute	Value
⊟ Wireless LAN Parameters	(...)
BSS Identifier	1
Access Point Functionality	Disabled
Physical Characteristics	Direct Sequence
Data Rate (bps)	5.5 Mbps
⊞ Channel Settings	Auto Assigned
Transmit Power (W)	0.003
Packet Reception-Power Threshold...	-95
Rts Threshold (bytes)	None
Fragmentation Threshold (bytes)	None
CTS-to-self Option	Enabled
Short Retry Limit	7
Long Retry Limit	4
AP Beacon Interval (secs)	0.02
Max Receive Lifetime (secs)	0.5
Buffer Size (bits)	256000
Roaming Capability	Enabled
Large Packet Processing	Drop
⊞ PCF Parameters	Disabled
⊞ HCF Parameters	Default
⊞ High Throughput Parameters	Default 802.11n Settings
⊞ WAVE Parameters	Not Supported

Figure 5.43 WLAN settings for all nodes

For the fixed sensing approach the customised MAC process model 'wlan_dispatch_cr_noIFS' was used, and three simulations were conducted with the sensing duration set to 1 ms, 50 ms and 100 ms. For the proposed QACR-MAC sensing strategy, the customised MAC process model 'wlan_dispatch_cr_noIFSPS4' was used. The QACR-MAC was implemented to select $S_d[AC]$ based on the AC of the frame, such as S_d[AC_VO] = 1 ms, S_d[AC_VI] = 5 ms, S_d[AC_BE] = 50 ms and S_d[AC_BK] = 100 ms, as shown in Table 5.6. The sensing was assumed perfect in all simulations.

Figure 5.44 illustrates the average delay, measured for each type of AC traffic in the network, when the 1 ms fixed sensing approach is used. After 460 seconds of simulation time, the average delay of voice and video traffic steadied at around 3 seconds, and for best effort and background traffic were around 3.8 seconds. Therefore, the IEEE 802.11e mechanism performed poorly: there is no noticeable difference between the different types of traffic. At 1 ms sensing, as shown in Figure 5.45, the differentiation was less than expected but there was somewhat decreased in overall average delay and stability compared to the 100 ms case. When the proposed solution QACR-MAC was used the priority mechanism of IEEE 802.11e was maintained with stable and low average delays for all traffic, as presented in Figure 5.46. In particular, for voice traffic, the average delay was stable at around 0.004 seconds from the beginning of the simulation time.

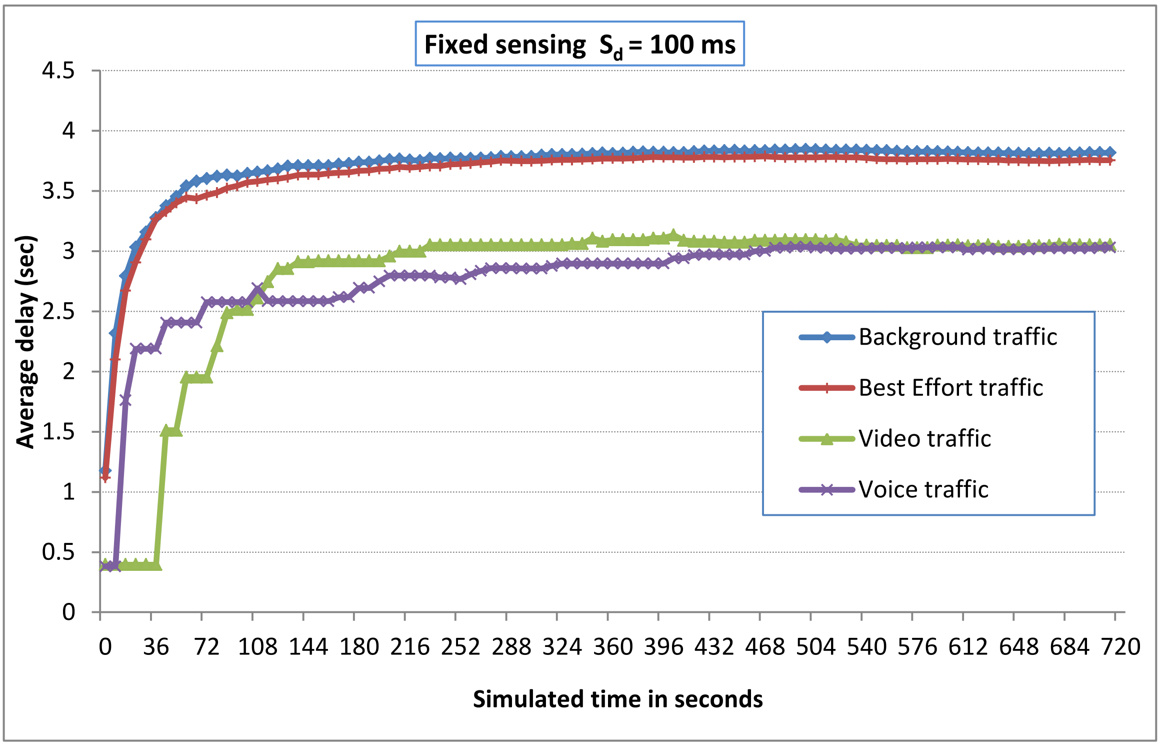


Figure 5.44 Average delay for each AC traffic when S_d = 100 ms fixed sensing is used

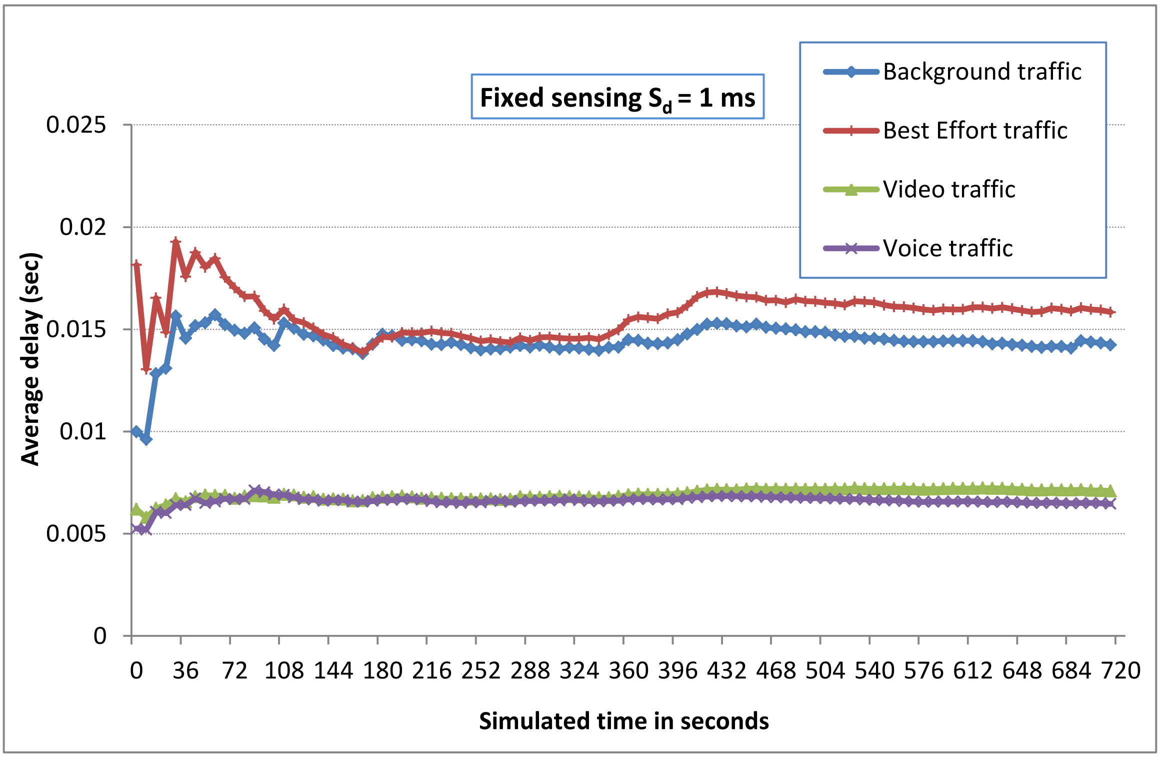


Figure 5.45 Average delay for each AC traffic when S_d = 1 ms fixed sensing is used

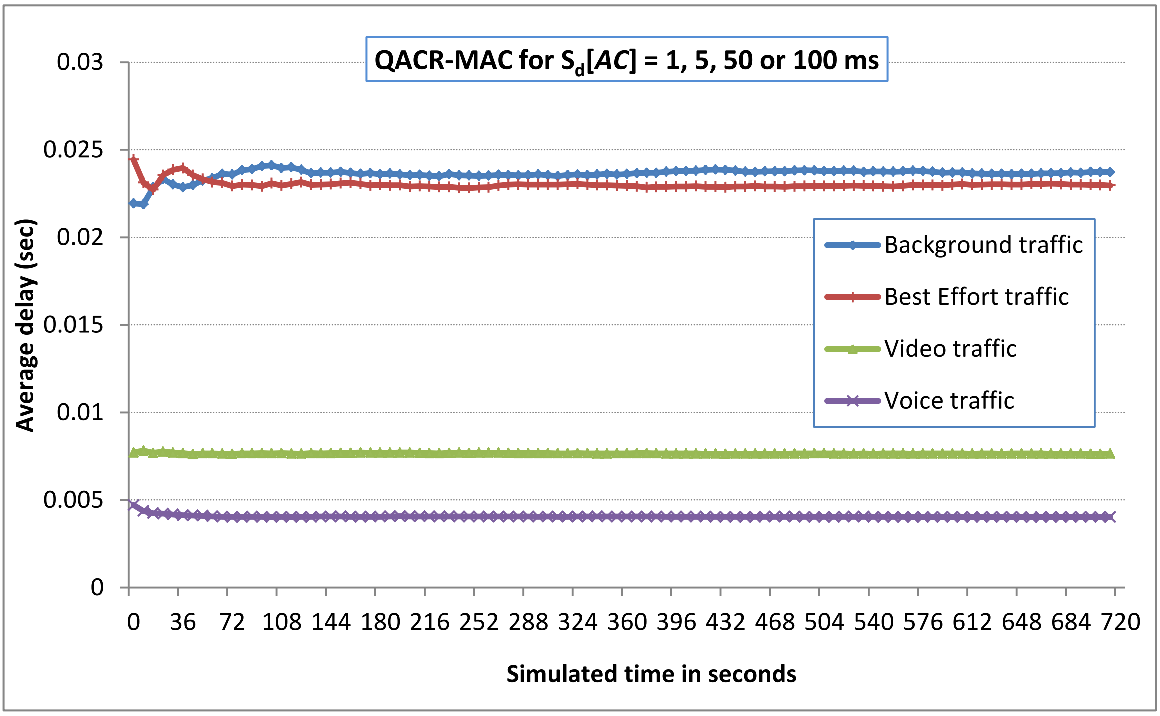


Figure 5.46 Average delay for each AC traffic when QACR-MAC sensing is used

5.5.3. Impact of frame aggregation and sensing

As discussed in Section 4.3.3, the used frame aggregation technique has an impact on sensing frequency, and consequently on PU protection. Aggregation will elongate the transmission frame to achieve higher throughput. As a result, sensing frequency is reduced and the possibility of interference with a PU increases. Two types of frame aggregation mechanisms, aggregated MAC service data unit (A-MSDU) and aggregation MAC protocol data unit (A-MPDU), are supported in IEEE 802.11 networks. In this section, various aggregation settings are studied under the use of the proposed sensing QACR-MAC. The Riverbed Modeler has a pre-setup scenario to study frame aggregation, called Frame_Aggregation_Study. This scenario was used to study the effect of enabling or disabling A-MSDU and A-MPDU on throughput and delay in a saturated network when QACR-MAC is used. The scenario layout is displayed in Figure 5.47.

The scenario consists of four APs each representing a different basic service set (BSS) with diverse frame aggregation configurations:

- BSS-0 (within blue circle): no frame aggregation is enabled.
- BSS-1 (within red circle): A-MSDU is enabled for traffic category ‘interactive multimedia’.
- BSS-2 (within green circle): A-MPDU is enabled for traffic category ‘interactive multimedia’.
- BSS-3 (within a light blue circle): A-MSDU and A-MPDU are enabled for traffic category ‘interactive multimedia’.

Each BSS has ten nodes connected to it with the same aggregation configurations. The naming scheme of these nodes follows the below pattern:

- BSS-0: Access Point_1 with connected node’s name goes from Mobile_1_1 to Mobile_1_10
- BSS-1: Access Point_2 with connected node’s name goes from Mobile_2_1 to Mobile_2_10
- BSS-2: Access Point_3 with connected node’s name goes from Mobile_3_1 to Mobile_3_10
- BSS-3: Access Point_4 with connected node’s name goes from Mobile_4_1 to Mobile_4_10

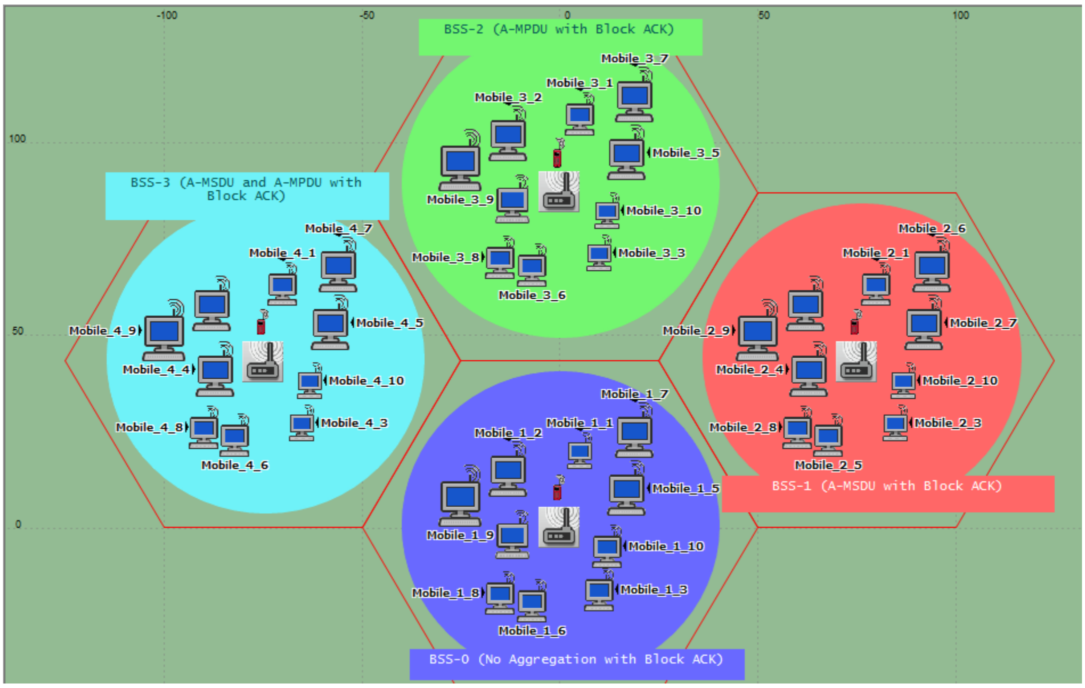


Figure 5.47 Network layout used for frame aggregation study (scale in meters)

Snapshots of the frame aggregation configurations for BSS-0, BSS-1, BSS-2 and BSS-3 are illustrated in Appendix B. The channels are configured such that there is no interference between any two BSS. For the sake of comparison, the relative position of a given node from the AP remains the same across all BSS. Each BSS has one jammer node configured to operate in the same frequency band: that is, Jammer_1 interferes only with transmissions in BSS-0, Jammer_2 interferes only with transmissions in BSS-1, Jammer_3 interferes only with transmissions in BSS-2 and Jammer_4 interferes only with transmissions in BSS-3. The jammer nodes are active from 20 seconds to 40 seconds during the total simulated time of 60 seconds. The jammers transmit packets of size 100 bits at 1 Mbps with an interval time of 0.005 seconds. The attribute settings of the four jammers are collected and displayed in Appendix B. More detail about the used traffic 'interactive multimedia' and the node attributes are available in Appendix B. The traffic is configured with 'interactive multimedia' types of service; hence it belongs to AC 'video'. By default, AC 'video' has a longer TXOP, a shorter AIFS time, and a shorter contention window than AC 'best effort': with an AC 'video' a node could send multiple higher layer packets in a TXOP. All nodes use B-ACK for the traffic category 'interactive multimedia'.

The maximum A-MSDU size that a node can transmit is set to the maximum allowed size, which is 7935 bytes in all nodes in BSS-1 and BSS-3 that are capable of transmitting an A-MSDU frame. 'RTS threshold' is set to 1000 bytes in all nodes in BSS-2 and BSS-3 that are capable of transmitting an A-MPDU. In BSS-3, when A-MSDU is carried in an MPDU as a part of A-MPDU, the maximum A-MSDU size is limited by the maximum MPDU size within an A-MPDU sub-frame, which is 4095 bytes. As in IEEE 802.11af the maximum transmission time, aPPDUMaxTime, allowed for an aggregated frame is 20 ms [155], the maximum possible inference to a PU should be 20 ms when perfect sensing is assumed. Potentially, the jammer interference in this scenario could be assumed from PUs or SUs belonging to other technologies.

To study QACR-MAC under different aggregation settings, all nodes were configured to use the customised MAC process model 'wlan_dispatch_cr_noIFSPS4'. In this scenario the QACR-MAC was implemented to select $S_d[AC]$ based on the AC of the frame, such as S_d[AC_VO] = 5 ms, S_d[AC_VI] = 10 ms, S_d[AC_BE] = 100 ms and S_d[AC_BK] = 300 ms. The S_d is increased for each AC, compared to previous sections, to study the frame aggregation under high sensing accuracy. Higher accuracy is required to improve PU protection that effected by the frame aggregation. The proposed sensing selection mechanism in QACR-MAC is still working for aggregated frames, as these frames must be belong to the same AC in both A-MSDU and A-MPDU. This helps to maintain the IEEE 802.11e QoS mechanism.

The measured throughput for video traffic for each AP per BSS is shown in Figure 5.48. The throughput was significantly affected on all BSS when the jammers were active from 21 seconds to 39 seconds, compared the throughput achieved when the jammers were inactive for 40 to 60 seconds. When the jammers were active, Access Point_3 in BSS-2 with A- MPDU enabled had the best throughput. In contrast, when the jammers were inactive, the Access Point_2, in BSS-1 with A-MSDU enabled had the best throughput. The delay experienced for video traffic for each AP per BSS is presented in Figure 5.49. As with the throughput observation, the delay noticeably increased in all BSS when jammers were active compared to when they were inactive. The delay was the smallest for BSS-2 with A-MPDU enabled when jammers were active, and least for BSS-1 with A-MSDU enabled when they were inactive. Table 5.7 summarises the comparisons between the BSS in terms of throughput and delay.

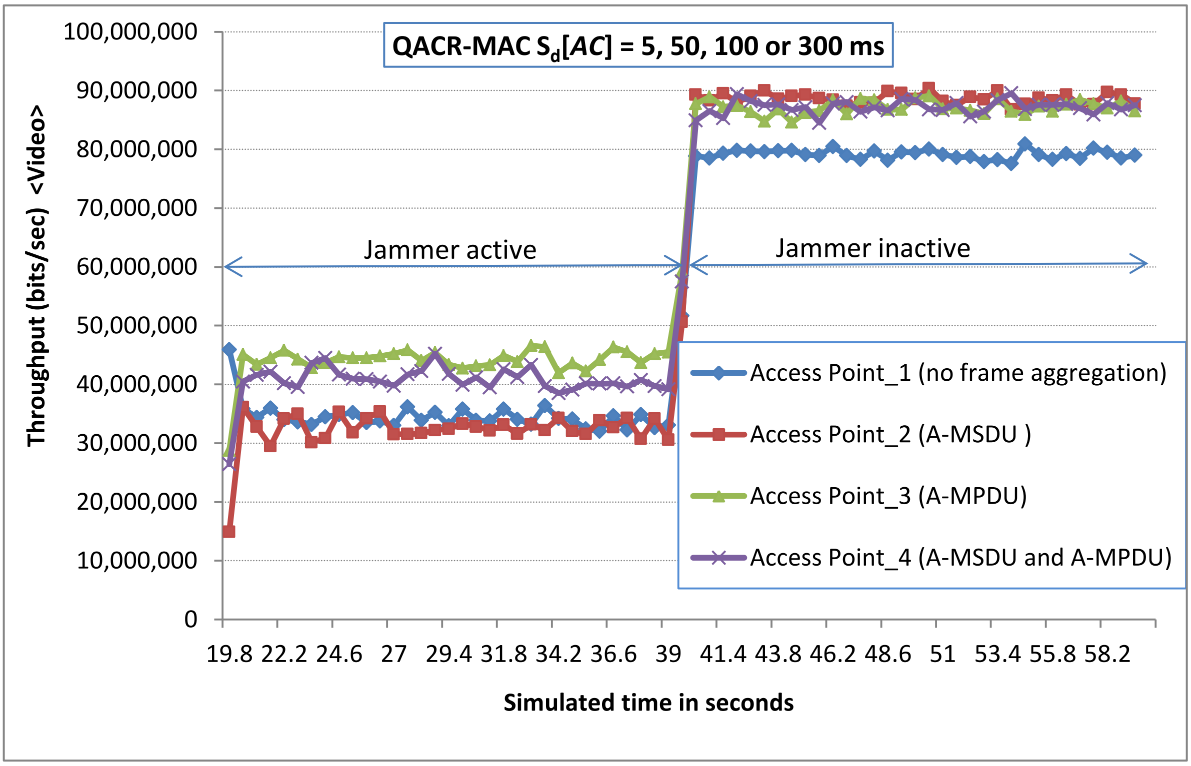


Figure 5.48 Throughput for video traffic under different frame aggregation configurations (QACR-MAC)

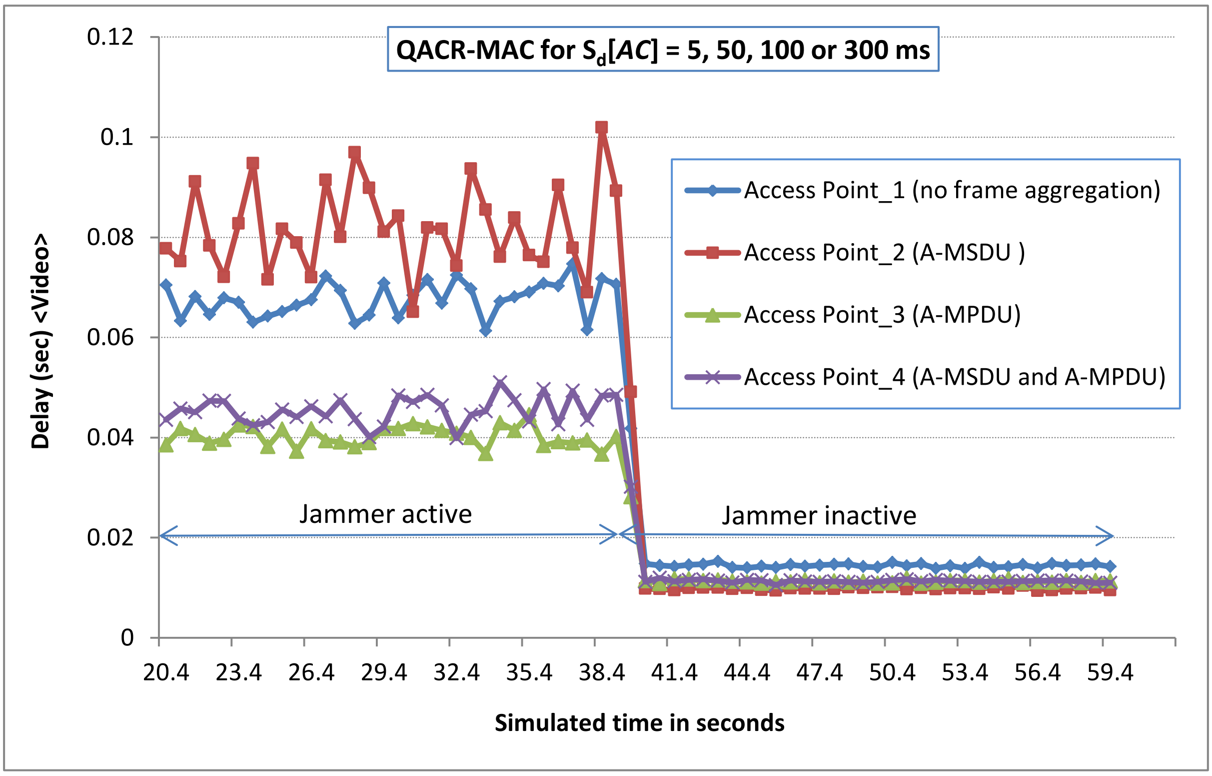


Figure 5.49 Delay for video traffic under different frame aggregation configurations (QACR-MAC)

Table 5.7 QoS comparison between frame aggregation mechanisms for QACR-MAC.

Aggregation mechanism	Throughput		Delay		Comments
	Jammer	No Jammer	Jammer	No Jammer	
No aggregation (BSS-0)	Third-Highest	Lowest	Second-Highest	Highest	Poor performance
A-MSDU enabled (BSS-1)	Lowest	Highest	Highest	Lowest	Best performance only in low errors channel condition
A-MPDU enabled (BSS-2)	Highest	Third-Highest	Lowest	Third-Highest	Best performance only in error channel condition
A-MSDU and A-MPDU enabled (BSS-3)	Second-Highest	Second-Highest	Third-Highest	Second-Highest	Good performance

The simulation results show that BSS-0 without frame aggregation performed poorly when the jammer was inactive and worse when it was active. When frame aggregation was disabled, the transmitted frames were smaller but with more overall overhead. The overhead included the information added at the MAC layer for each frame, such as MAC addresses and preambles, and the sensing duration required before sending each frame. Therefore, sensing was conducted more frequently—and that may not be necessary, particularly for high-throughput channels with low protection requirements for their PUs.

The frame aggregation mechanism allowed nodes to send more data within the given TXOP by aggregating frames in one frame to be sent, using the transmission channel more efficiently. The sensing could be conducted less frequently than in the case of transmission without frame aggregation. As long as it complies with the regulations protecting PU transmission, the less frequent sensing helps reduce sensing overheads, achieving higher QoS. However, that will depend on which mechanism is used for aggregation, and under what conditions, as the results in this section show.

To understand roughly the number of data packets sent per aggregated frame, called physical protocol data unit (PPDU), samples of statistics from one node in each BSS are presented in Figure 5.50, Figure 5.51, Figure 5.52 and Figure 5.53. In BSS-1 the A-MSDU was enabled, so larger aggregated frames were formed for video packets with less MAC overhead. The MAC overhead reduced as the MAC header and cyclic redundancy check added only to the whole A-MSDU and not to its sub-frames, as in A-MPDU. The number of MSDUs that were packed into an A-MSDU can be observed in Figure 5.50. The number of MSDUs aggregated in an A-MSDU was higher in Mobile_2_1 than Mobile_4_1 (compare Figure 5.52 with Figure 5.50). Although A-MSDU was enabled in Mobile_4_1, the maximum A-MSDU size is limited by the maximum MPDU size within an A-MPDU sub-frame, which is 4095 bytes, because A-MPDU was also enabled. Therefore, BSS-1 with A-MSDU has the fewest sensing and MAC overheads. For example, Mobile_2_1 in BSS-1 had the lowest media access delay compared to nodes belonging to the other BSS, as shown in Figure 5.54. As a result, A-MSDU enabled the BSS-1 network to use the available channel more efficiently, and achieved the highest QoS when the channel condition was good (when the jammer was inactive).

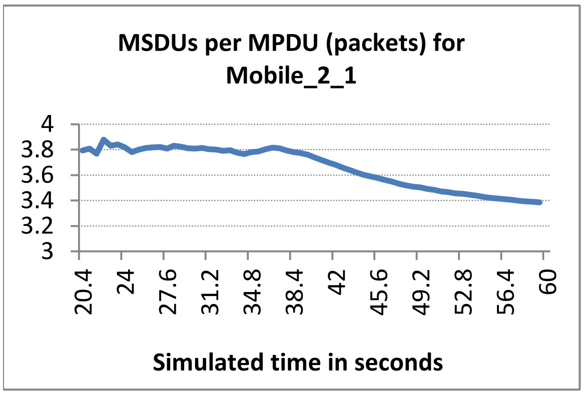


Figure 5.50 Number of aggregated MSDUs per MPDU for Mobile_2_1 (BSS-1)

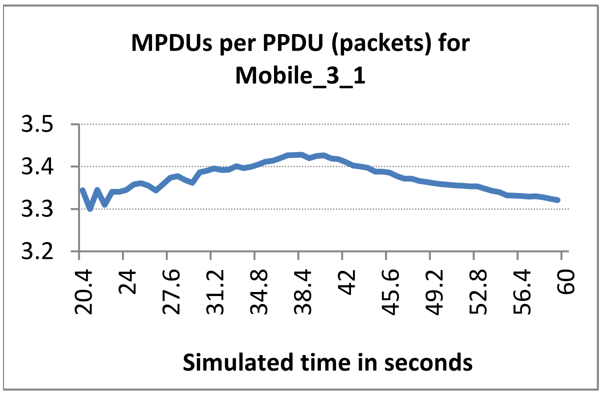


Figure 5.51 Number of aggregated MPDUs per PPDU for Mobile_3_1 (BSS-2)

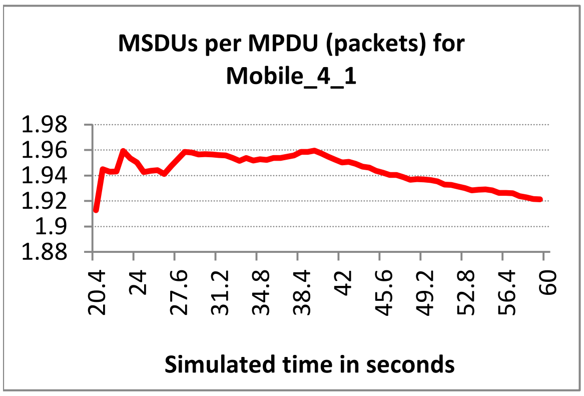


Figure 5.52 Number of aggregated MSDUs per MPDU for Mobile_4_1 (BSS-3)

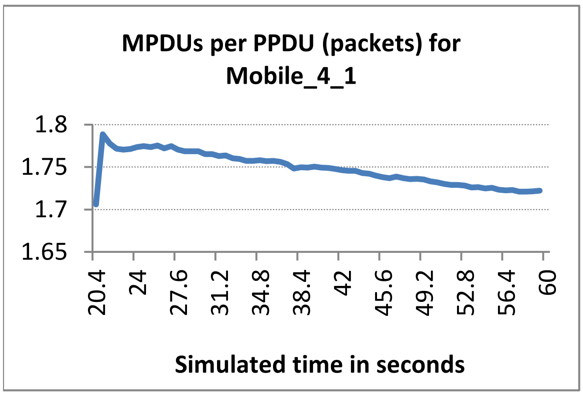


Figure 5.53 Number of aggregated MPDUs per PPDU for Mobile_4_1 (BSS-3)

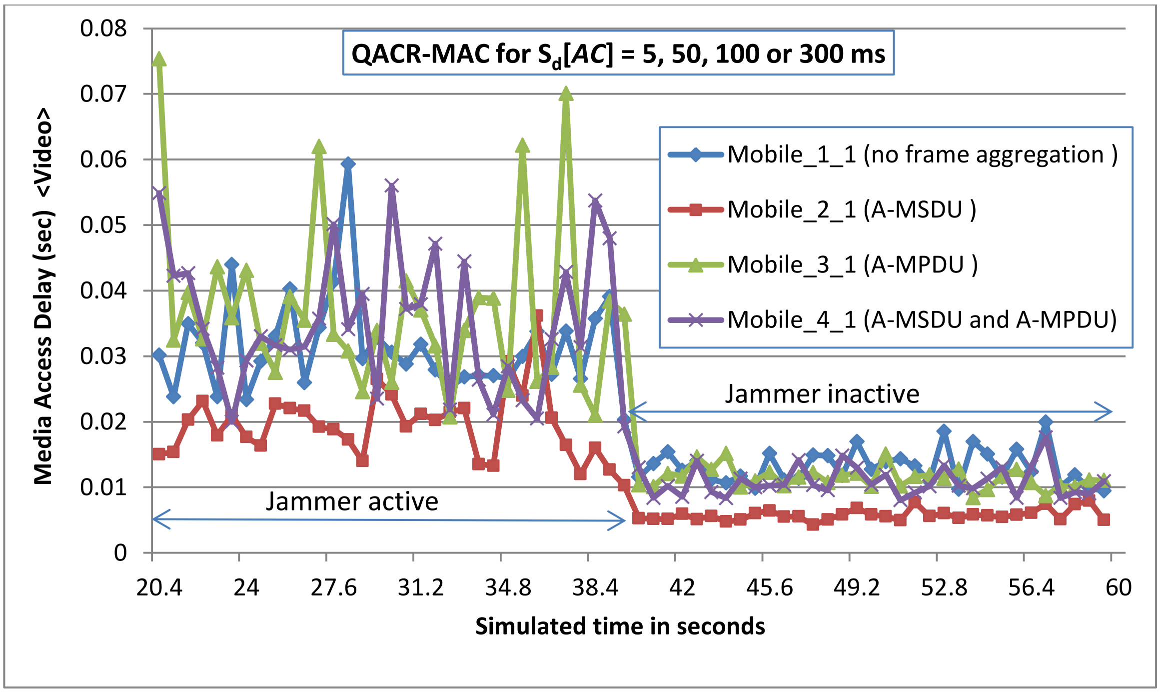


Figure 5.54 Media access delay for video traffic under different aggregation configurations (QACR-MAC)

The QoS, in terms of throughput and delay, was worst when the jammer was active, as it caused more transmission errors in the operational channel because of its interference. In A-MSDU, if one sub-frame is corrupted, the whole A-MSDU has to be retransmitted, so the A-MSDU mechanism performed poorly when the jammer was active.

In BSS-3, both A-MSDU and A-MPDU were enabled. The resulting aggregation mechanism is a kind of compromise between A-MSDU and A-MPDU techniques. In this case, each resulting PPDU contains aggregated MPDUs with aggregated MSDUs inside them. A sample of the statistics of MSDUs per MPDU for Mobile_4_1 is shown in Figure 5.52, and the number of those MPDUs per PPDU for the same node is shown in Figure 5.53. The results show that BSS-3 performed well whether its jammer was active or inactive. While the throughput when the jammers were inactive was not high as BSS-1 because of the MAC overhead of aggregated MPDUs, it was not affected by the active jammer as much as BSS-1 because the maximum A-MSDU size was limited to the maximum MPDU size within an A-MPDU sub-frame, which is 4095 bytes. While in BSS-1, the A-MSDU maximum size was 7935 bytes (see snapshots of frame aggregation configurations in Appendix B).

5.5.4. Impact of coexistence

One of the challenges facing CR networks is how to handle different SU systems in the same spectrum holes. In particular, it is about how to avoid or reduce inference between the coexisting systems. Interference under the CR concept may accrue between PUs and SUs or between SUs. While it is compulsory in CR networks to avoid interference with PUs and leave the channel for them, SUs belonging to different systems may share the same spectrum hole as long as they can handle the interference between them. Spectrum sensing is important in distinguishing between PU and other SU transmissions.

In this section, the impact of the coexistence of WLAN nodes using the proposed QACR-MAC with ZigBee nodes was considered. A coexistent WLAN-ZigBee scenario implemented in Riverbed Modeler was adopted to study the impact of interference between WLAN and ZigBee networks. The network layout for studying their coexistence is displayed in Figure 5.55. The WLAN network consists of two mobile nodes, Mobile_1 and Mobile_2, and snapshots of the attributes of these are aligned in Figure 5.56. The mobile nodes were implemented with the MAC process model ‘wlan_dispatch_ cr_noIFSPCH’ to run simulations for a fixed sensing approach when S_d = 5 ms, 10 ms and 100 ms, then were configured to use the MAC process model ‘wlan_dispatch_cr_noIFSPS4’ to use QACR-MAC with the sensing selection settings shown in Table 5.8. The ZigBee network consists of a coordinator and an end device using the standard IEEE 802.15.4 MAC protocol. Snapshots of the attributes of the ZigBee nodes are found in Appendix B. The nodes were configured to use overlapping channels. The mobile nodes were assigned the same trajectory to move across the ZigBee network transmission during the simulation. The simulated time was 240 seconds and the two networks were closest to each other around the middle of the simulated time; i.e., around 130 seconds. The two networks at the beginning and towards the end of the simulated time were far enough from each other to avoid any interference between them.

Each network was configured to generate data traffic from higher layers at one node. The load traffic (best effort) generated from a higher layer and submitted to the MAC layer at Mobile_1 is shown in Figure 5.57. The load traffic from the higher layer and submitted to the MAC layer at the end device is can be found in Appendix B. The same load traffic was used in all simulation runs.

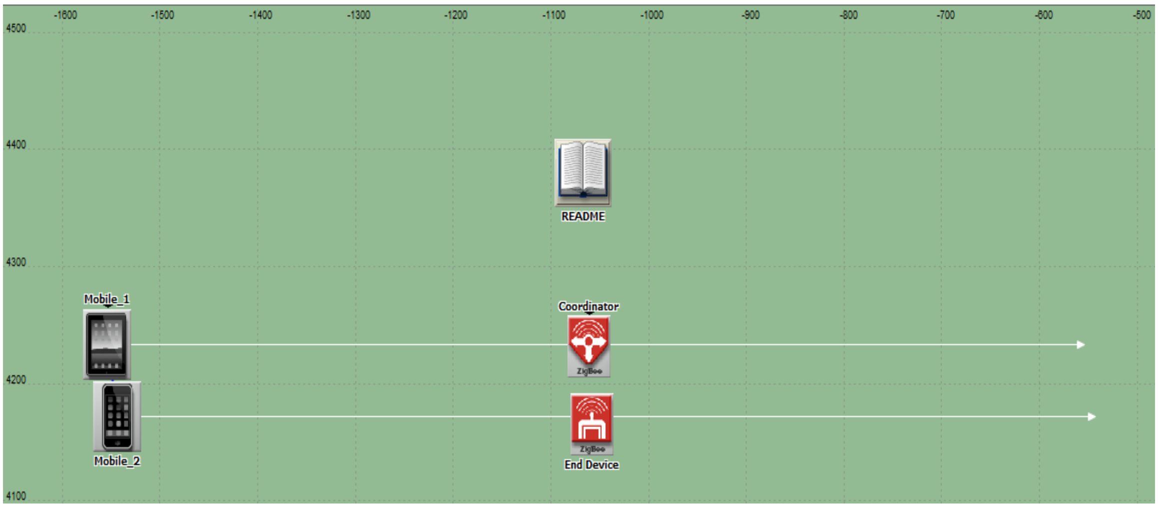


Figure 5.55 Network layout for studying WLAN-ZigBee coexistence (scale in meters)

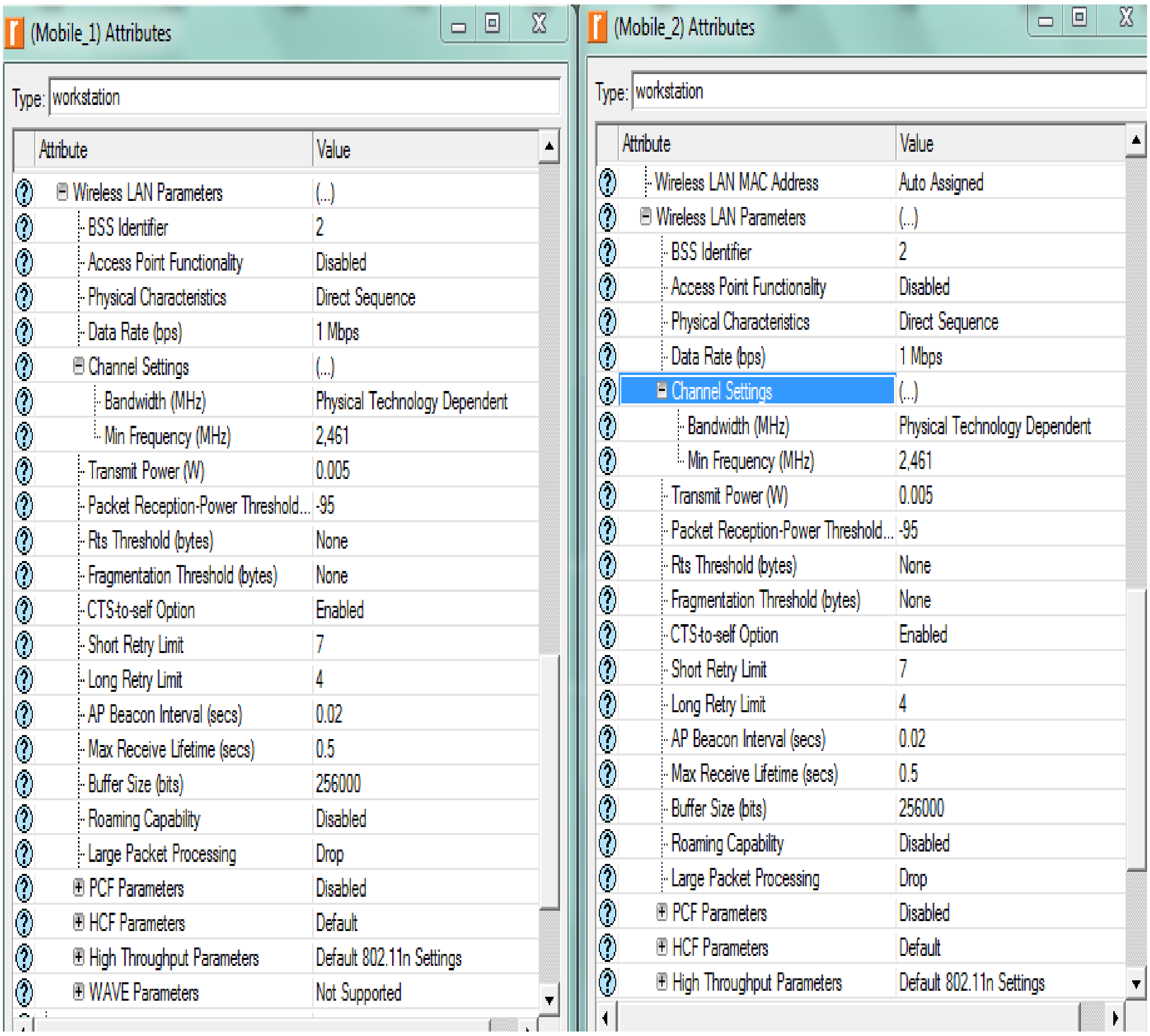
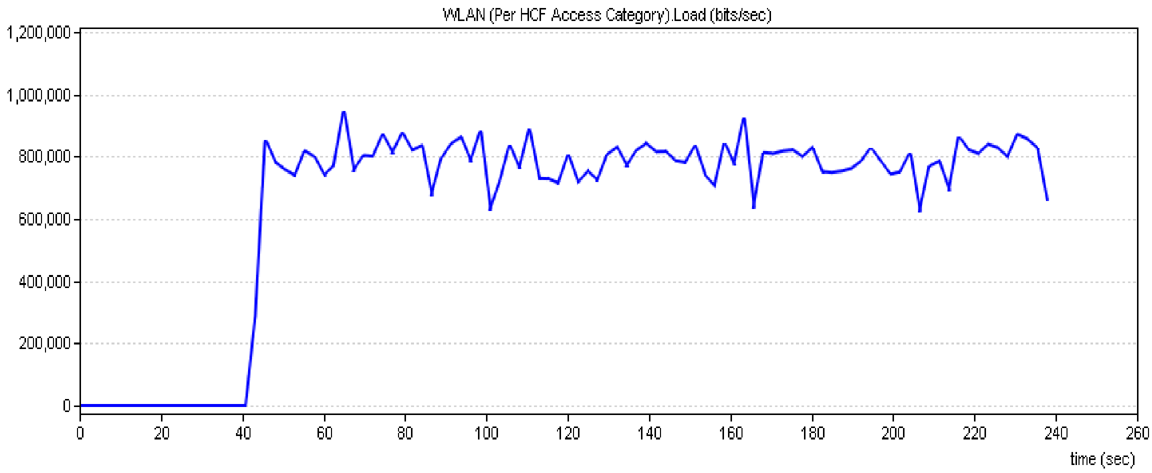


Figure 5.56 Attributes of mobile nodes



Simulated time in seconds

Figure 5.57 Load traffic from Mobile_1 to Mobile_2 (best effort)

Table 5.8 The imperfect sensing duration for each AC in QACR-MAC

Frame Access Categories	Sensing mode	Sensing duration (S_p) in milliseconds (ms) with imperfect sensing (P_d, P_f)
Voice (AC_VO)	Coarse sensing	S_d[AC_VO] = 1 ms with **($P_d = 0.9$, $P_f = 0.1$)**
Video (AC_VI)	Moderate sensing	S_d[AC_VI] = 5 ms with **($P_d = 0.95$, $P_f = 0.1$)**
Best effort (AC_BE)	Fine sensing	S_d[AC_BE] = 50 ms with **($P_d = 0.95$, $P_f = 0.1$)**
Background (AC_BK)	Extra Fine sensing	S_d[AC_BK] = 100 ms with **($P_d = 0.95$, $P_f = 0.01$)**

The throughputs achieved for best traffic in Mobile_2 under the different simulated sensing strategies are shown in Figure 5.58. From 90 up to 180 seconds of the simulated time, the throughput was significantly affected by ZigBee transmission interference as the two networks came close to each other. The results show that the QACR-MAC approach achieved a higher throughput than the fixed sensing approach even when the sensing duration was 5 ms.

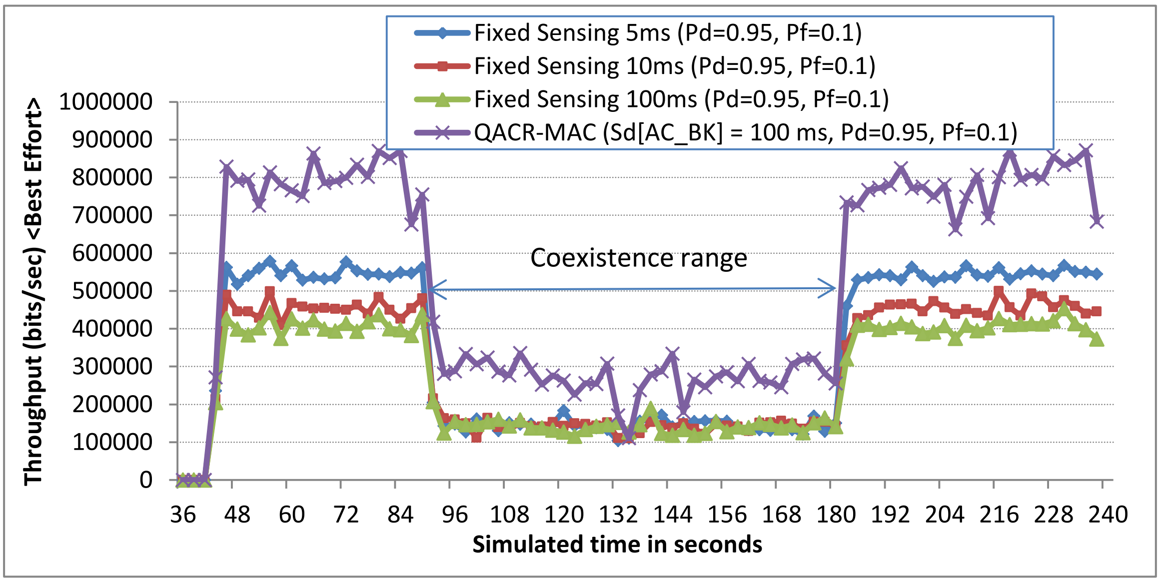


Figure 5.58 The throughput at Mobile_2 under different sensing strategies for coexistence scenario

The QACR-MAC mostly used the 100 ms sensing duration with P_d = 0.95 and P_f = 0.1 because the generated traffic AC was the best effort. However, the proposed strategy QACR-MAC can achieve higher throughput than the fixed sensing approach with a shorter sensing duration; that is, of 5 ms, under the same accuracy. In reality, in shorter sensing times high-accuracy methods cannot be used and lower accuracy is expected. The findings prove that QACR-MAC presents a noticeable improvement in its throughput by reducing the sensing overhead while providing higher accuracy. The results also point out the issue of handling the possible coexistence of different SU standards. Regarding this issue, as discussed in Section 4.2, spectrum sensing faces the challenge of distinguishing between a PU transmission and other SU transmissions.

5.5.5. Impact of CR nodes number

This section considers the impact of increasing SU nodes on the delay in a CR network when QACR-MAC is used. For this study, an ad-hoc network was created with 21 nodes, one of them configured as a video server to generate video traffic. In the first simulation run only the server and another two nodes were activated, as shown in Figure 5.59. Several other simulations were run, in each the number of activated nodes increasing by one until all nodes were active (see Appendix B). The applications and profiles nodes were used to define video conferencing with light traffic and its profile 'Video' as shown in Figure 5.60 and Figure 5.61.

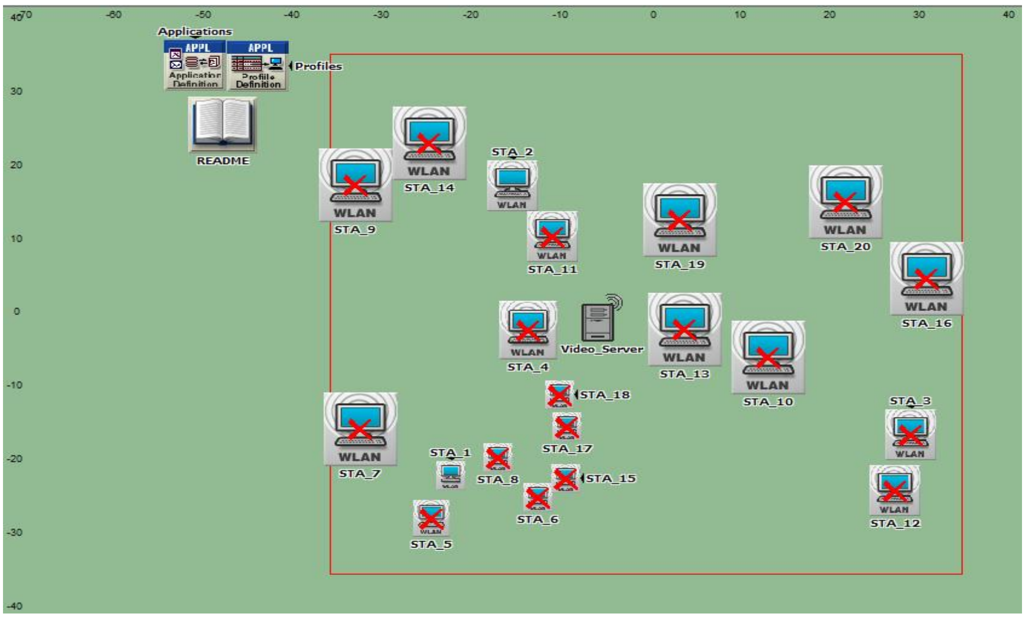


Figure 5.59 Network layout for studying the CR nodes number impact (scale in meters)

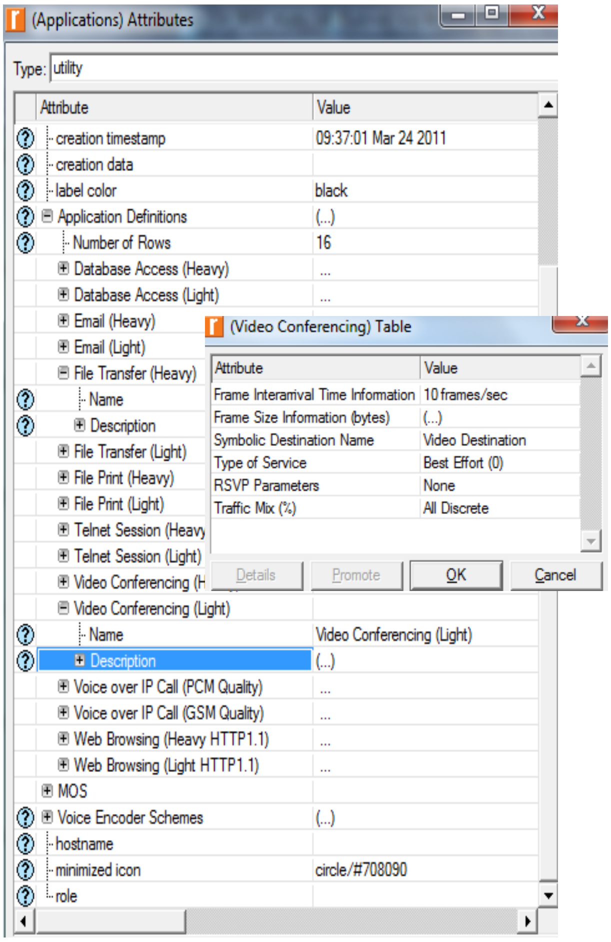


Figure 5.60 Application video conferencing

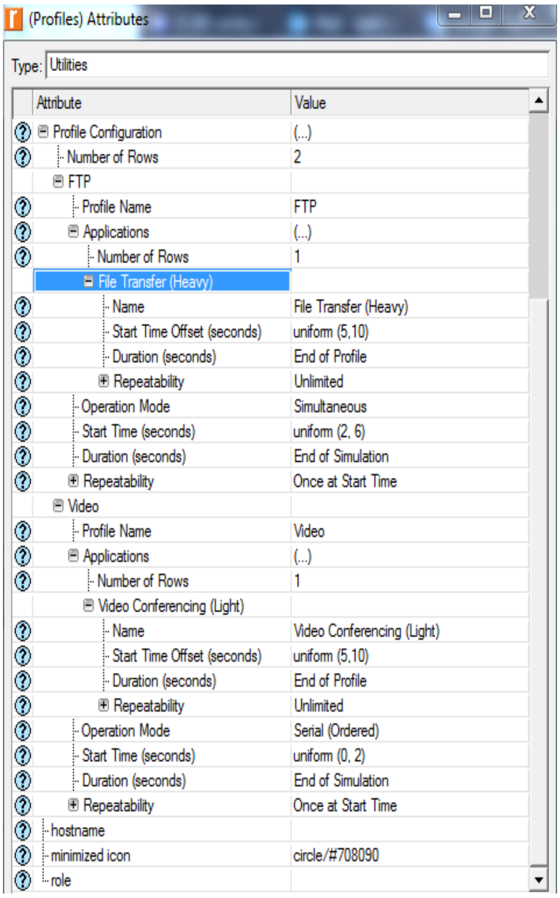


Figure 5.61 Profile configurations for Video

Video_Server was configured as a video conferencing server, and other nodes with the profile 'video', to use the server for exchanging video traffic. The WLAN settings for all nodes are displayed in Appendix B. All nodes were implemented to use MAC process model 'wlan_dispatch_cr_noIFSPS4' for QACR-MAC sensing strategy with the settings shown in Table 5.8. The AC of the video traffic was set to 'best effort' (see Figure 5.60, the small settings window) so the sensing duration 100 ms was mostly selected by QACR-MAC. The simulated time was 30 seconds for each run, and the other simulation parameters were as in Table 5.5.

Figure 5.62 compares the overall averages delays measured at the Video_Server when the number of active nodes differs. Generally, delay increased as the number of nodes increased, but there was an exception when the number of active nodes was 19: the delay here was lower than for 17 nodes, but higher than for 15 nodes. Such a finding is reasonable in a random access environment.

The maximum average delay was around 0.016 seconds for 21 nodes, which is acceptable for most voice and video applications. Several previous proposed solutions based on the fixed sensing approach experienced a higher average delay, e.g., more than 0.2 seconds, for the same number of nodes, as reported in [119, 144, 210]. The packet delay variations measured at STA_2 when the number of nodes was 3, 7, 11 and 19 along the simulated time is illustrated in Figure 5.63. The delay variations increased as the number of nodes increased. The measured packet delay variation was much lower, even for 19 nodes, than that measured for the fixed sensing approach reported in Figure 5.32.

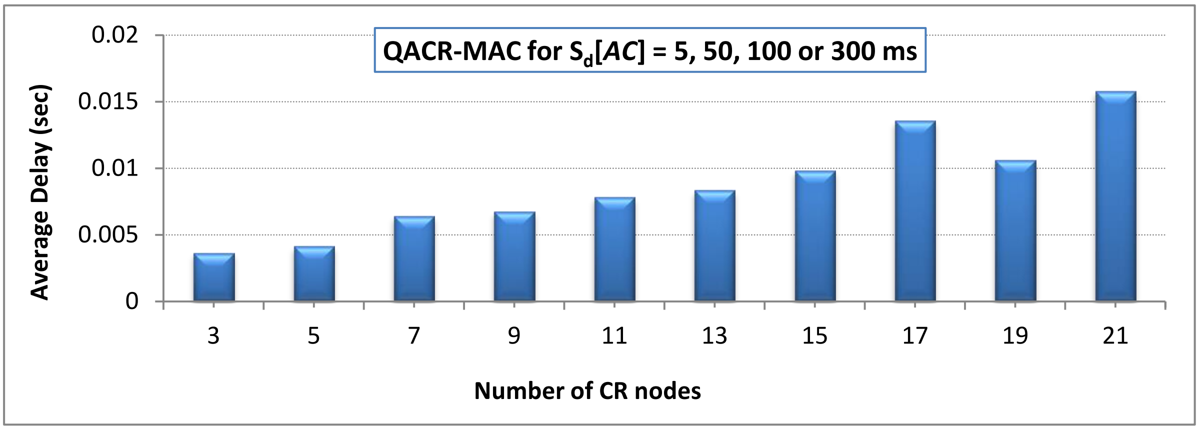


Figure 5.62 Comparing average delay in Video_Server based on the number of nodes

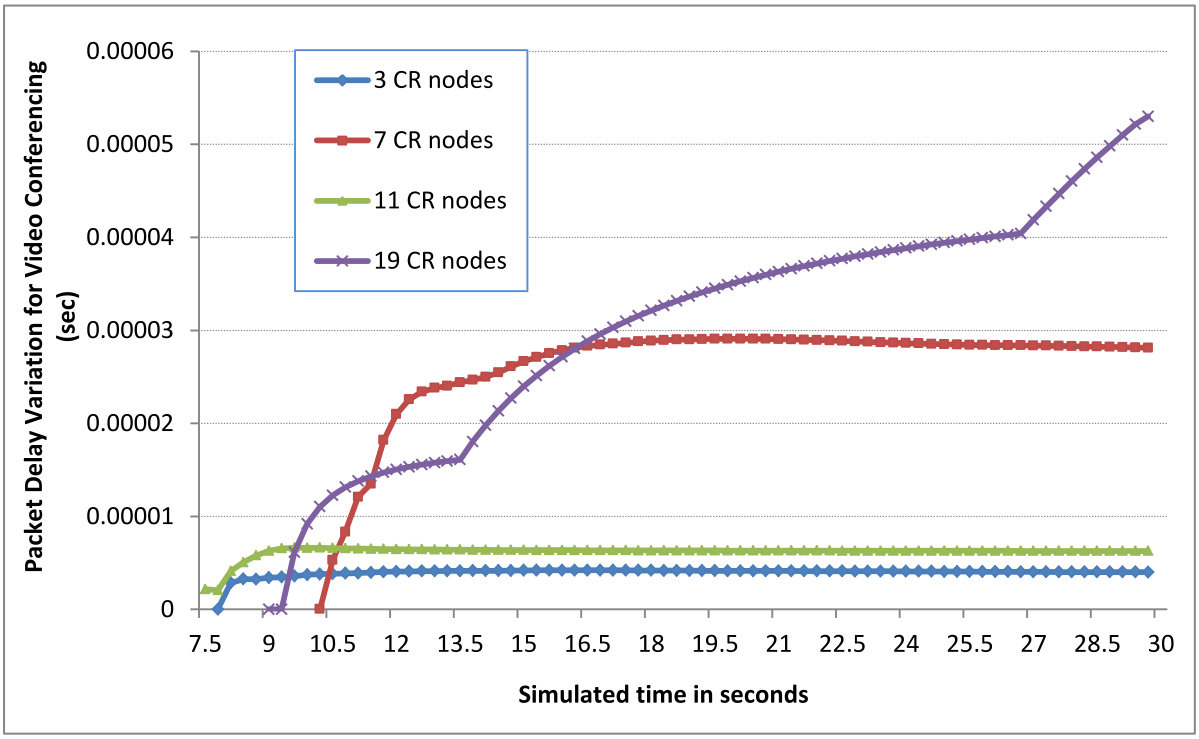


Figure 5.63 The packet delay variation in STA_2 for different number of activated CR nodes (video traffic)

5.6. Summary and conclusions

In this chapter the wide range of simulations and their results provide valuable resources to understand QoS issues in a CR network related to its sensing operations and how to address them. In this chapter, the solutions proposed in Chapter 3 and particularly Chapter 4 are implemented and evaluated. The simulation tool Riverbed Modeler (OPNET) provides a range of features to study wireless technologies and variously related statistics during different simulations. The CR networks and spectrum sensing accuracy parameters have not been implemented yet in the Modeler. The features and limitations of the simulation tools used in this study are also considered. Taking into account sensing accuracy, different sensing strategies are implemented for IEEE 80211 nodes in the simulation tool, and their features and limitations are considered.

Four main sensing strategies are implemented; they have been named extreme sensing, fixed sensing, select sensing and QACR-MAC. The first two strategies represent conventional strategic approaches that can be found in the literature, as discussed in Chapter 3. The select sensing strategy represents the use of the fuzzy logic selection mechanism to select the proper sensing method based on given factors, as explained and implemented by using MATLAB in Chapter 3. The select sensing strategy uses the selection mechanism before sending any frame, which is costly in random access networks but still performs better than conventional sensing strategies, as the results in this chapter show. The QACR-MAC strategy refers to the solution proposed in Chapter 4. In QACR-MAC, the proposed fuzzy logic selection mechanism is used in an efficient way that suits the random access MAC protocols used in IEEE 802.11.

The QoS issues in conventional sensing strategies are evaluated in Section 5.3 and Section 5.4. Particularly, the impact of sensing duration and imperfect sensing on different types of delay and throughput are analysed. The simulation results indicate significant QoS degradation when the sensing duration is increased for better accuracy. Hence, using high-accuracy sensing methods results in very low QoS in the fixed sensing strategy. Those sensing methods able to identify PU signals and reach high accuracy in low SNR required a long sensing duration. This finding and the characteristics of other sensing methods are discussed in detail in Chapter 2.

In Section 5.5, the proposed select sensing and QACR-MAC strategies are evaluated and compared with the fixed sensing strategy. During this study, no select sensing strategy considering the application requirements, either proposed or evaluated, was found in the literature.

The results show noticeable improvement, in particular, for QACR-MAC. The QACR-MAC strategy allows the selective use of longer sensing times to achieve higher accuracy with less impact on QoS metrics such as delay. The results also demonstrate the expected poor performance of IEEE 802.11e QoS mechanism when a fixed sensing strategy is used. In contrast, QACR-MAC achieves better QoS in White-Fi networks based on sensing and maintains the expected behaviour of IEEE 802.11e under different frame aggregation mechanisms. The overall results of using QACR-MAC strategy prove that it has the potential to improve the sensing functions and QoS in CR networks.

Chapter 6. Conclusion and Future work

This chapter summarises the conclusions of this dissertation and highlights potential future work. Section 6.1 summarises cognitive radio technology within the context of this study. The literature review regarding CR sensing function and its impact on QoS is concluded in Section 6.2. The proposed fuzzy logic decision-making scheme to select the appropriate sensing method is summarised in Section 6.2. The features of the sensing strategy proposed for White-Fi networks to improve sensing accuracy and QoS is briefly described in Section 6.4. The simulation work and results of analysing the sensing impact on QoS and evaluating the proposed solutions are concluded in Section 6.5, and the limitations and future work are found in Section 6.6.

6.1. Cognitive radio

The cognitive radio (CR) concept introduces innovative approaches to using the radio frequency spectrum (RFS) and implementing wireless systems. Theoretically, the term 'CR' implies that wireless devices and systems are capable of perceiving the surrounding RFS conditions. CR systems can adapt to identify RFS situations to achieve better performance and spectrum utilisation, based on two related technologies: software defined radio (SDR) and dynamic spectrum access (DSA). Principally, CR capabilities are provided by a progression of functions, called a CR cycle, starting with the spectrum sensing function and managed by the spectrum decision and mobility functions, then acting accordingly by executing a spectrum sharing function. The concept of CR is applied to current RFS regulations to establish various technologies that may help to address the scarcity and underutilisation of useful RFS bands. The first step is to allow devices with CR capabilities, called CR devices, CR users or secondary users (SUs), to exploit any vacant licensed RFS bands, known as spectrum holes or white spaces. The opportunistic use of such spectrum holes is conditional on not harming the licensed users, known as primary users (PUs). At this early stage, the focus is on the available spectrum holes in TV bands, also known as TV white spaces, because these low-frequency bands have desirable transmission characteristics. More importantly, the frequency, location and time of TV white spaces, can be forecast, and offered in advance to SUs through a geolocation database (GDB). In this case, cognition capability is based on information provided from outside the CR device. More resources are required, such as a communicating channel with GDB and global positioning (GPS) facilities. Projecting PU activities may not be possible in other RFS bands, so the GDB approach cannot be generalised, or even considered a reliable solution. An SU must be able to recognise the surrounding RFS based on itself, by using a spectrum sensing approach. Ultimately the outcome of spectrum sensing in CR is a crucial information source used by the other CR functions, and therefore its accuracy has a core role in utilising RFS. Typically this function is conducted by periodically using one of the available spectrum sensing techniques during CR operation. Practically, the sensing operation shares the transmission resources, such as the time for transmission, with actual data transmission. Consequently, the throughput and latency for the data communication will be affected by sensing. Such sensing overheads, especially on the QoS, are among the key challenges in targeting other RFS spectrum holes, rather than TV white spaces, where GDB may not be visible. This study concentrates on addressing the impact of spectrum sensing on QoS, as the wide adoption of CR technologies depends on how well this issue is resolved.

6.2. Sensing function and QoS

Regarding CR, spectrum sensing may be local or cooperative. Local sensing is conducted by a CR device using one of the available sensing techniques, while cooperative approach is based on sharing information gathered by local sensing among SUs, via one of the local sensing techniques. These local sensing techniques mainly involve blind sensing or building on prior information, also known as signal specific sensing. Fundamentally, blind sensing attempts to determine if the channel is busy or idle, and does not rely on pre-known information about PU signals. In contrast, the other techniques based on prior information are able to identify a specific PU transmission among other transmitted signals or noise. The features and limitations identified in Chapter 2 for a range of sensing techniques indicate that none of them suits all CR operation requirements. For instance, blind techniques require less time and resources for their operation but have limited outcomes and accuracy. Conversely, sensing techniques with more accurate and advanced knowledge about RFS require higher resources and prior information about the sensed signals. In this study, three possible levels of sensing outcomes are identified. The basic and first level is for identifying if the channel is busy or idle. Such an outcome is not adequate for CR as it cannot tell if a channel is busy because of PU transmission or not. Blind sensing techniques such as the common one, energy detection (ED), do provide this necessary information although the usual accuracy parameters, including the probability of detection (P_d) and of false alarm (P_f), do not reflect accuracy in detecting PU signals as is commonly assumed in several proposed solutions based on this method. In the second level of possible outcomes, the sensing technique used is capable of distinguishing PU signals from among other transmissions. Third-level outcomes refer to the ability to recognise various signals as belonging to the PU and other SUs sharing the channel. In the future, more levels may be added to identify specific information that can be gained by assessing channels. For instance, spectrum holes may be identified based on unused codes of PU transmissions in Code-division multiple access (CDMA) systems. Although second and third levels help an SU make more informed decisions about using RFS and protecting PU signals, they involve more complex sensing techniques with a higher impact on QoS, requiring a trade-off between efficient sensing and QoS. How ongoing research will address this trade-off will depend on how sensing methods are used. Two main approaches are found and discussed in Section 2.5. The first is the more dominant in the literature. This thesis named this one 'fixed sensing strategy' as it does not change dynamically in response to operational requirements. It is designed either on a single sensing technique or on more than one, used in serial or parallel form. The second approach has a dynamic strategy of utilising a set of sensing methods by selecting the most proper one based on real-time changes in operational requirement, and it is named 'select sensing strategy' in this work. A few potential solutions have been based on this strategy, but with no clear awareness of QoS requirements. The select sensing strategy is expected to perform better than the fixed strategy and it did so in the simulation part of this work. Therefore, it was adopted for the study of the enhancement of QoS in CR networks.

6.3. Fuzzy logic scheme for selecting sensing methods

For an effective but simple selection mechanism, fuzzy logic algorithms were used to design and implement a novel decision-making model to select the most appropriate sensing method, as discussed in Chapter 3. The proposed selection mechanism requires only simple mathematical operation, so minimal overhead on the computational CR resources is achieved. Because of these features, the proposed solution is suitable for implementing in future networks, such as the Internet of Things (IoT), where wireless devices with varying capabilities are typically involved, running diverse applications. MATLAB was used to implement the proposed selection mechanism based on Mamdani's fuzzy inference system (FIS) as discussed in Chapter 3 and reported in Appendix A. The implemented FIS system consists of four fuzzy inputs, seventy three If-then rules and one output of four variable membership functions (MFs) representing four possible sensing methods. The FIS inputs denote the selection criteria, of four essential factors:

1. QoS requirements for the application running on the SU →'application QoS';
2. Protection level required for the PU transmission of the sensed channel →'PU protection';
3. Prior information available about the PU signal of the sensed channel →'prior information';
4. CR device capabilities of conducting the available sensing methods →'CR capability'.

These four factors were selected to consider the most important requirements and constraints in selecting a sensing method during operation. Limiting the number to four simplifies the system, but the capability to consider other factors is retained. The input 'application QoS' variables represent the various possible requirements of the upper layer, QoS requirements in particular. This factor reflects the cross-layered design, where any internal requirement may be considered. The 'PU protection' input variable may consider requirements from outside the system, such as those related to regulations protecting coexisting SU networks in specific frequency bands. The 'prior information' input may include any other constraints related to using the available sensing methods, such as the SNR of the sensed channel if it can be measured. Other constraints related to the CR device and network capabilities may be involved in this variable, such as remaining power resources. The FIS input and output values are fuzzy linguistic terms that can be adjusted with the proper MF for each value. Also, the FIS offers an easy and flexible way of tuning inputs, rules and outputs according to design goals.

The four sensing methods were designed as four possible values of the FIS single output without overlapping MFs. The four sensing methods were labelled from one to four instead of classifying them with specific sensing technique names, for more flexibility. Sensing method 1 represents the highest sensing technique from the available range in terms of outcome level and accuracy, which could be matched filter detection. Method 4 represents the lowest possible selected technique, typically ED. Method 1 has the highest impact on QoS, mainly because of the long duration used for sensing, while method 4 causes the lowest QoS degradation, especially in latency-sensitive applications. The FIS output also may be designed to use fewer than four sensing techniques but with different settings. For instance, methods 1 and 2 could be set for the same sensing technique while the other two are set to another technique, and all with different durations.

Based on the outcomes of the investigations in this work, the FIS If–then rules were defined. These rules facilitate the analytical selection of a sensing method that takes given requirements and constraints into account. The resulting decision-making selection scheme can be employed in different medium access control (MAC) protocols. The efficient sensing strategy, of when to use this selection algorithm and how frequent, may differ according to the MAC protocol used. In this study, the proposed sensing selection mechanism was applied to the de facto random access protocol, carrier sense multiple access with collision avoidance (CSMA/CA) for IEEE 802.11 wireless networks.

6.4. QoS-aware MAC protocol for sensing strategy

In Chapter 4, a sensing strategy called a QoS awareness CR MAC protocol (QACR-MAC), was proposed to integrate with the IEEE 802.11e QoS mechanism to enhance QoS in a CR network based on IEEE 802.11 standards. The IEEE 802.11af standard for White-Fi has, so far, been designed mainly to assess the spectrum based on the GDB. The challenges of using a spectrum sensing approach instead have not yet been addressed. For instance, the current 802.11af MAC protocols have not been designed to handle high-accuracy sensing with long duration while considering QoS. To address such challenges, the QACR-MAC proposed in this thesis aims to utilise high-accuracy sensing methods based on the sensing selection solution proposed in Chapter 4, with minimum impact on QoS. In QACR-MAC, the sensing selection is performed in the first attempt to transmit a data frame. The selection algorithm is initiated when a collision is deduced and the randomly chosen back-off time is adequate. As a result, high-accuracy sensing is achieved with minimal overhead. As the selection algorithm determines the sensing duration based on the QoS requirements categorised by IEEE 802.11e, this proposed solution integrates with the IEEE 802.11e mechanism to enhance overall QoS in White-Fi based on spectrum sensing.

The major concern about the current IEEE 802.11af is that is relies only on ED sensing that may trigger unnecessary handoffs or cause severe interference to a PU because it is limited and imprecise. By using QACR-MAC, the overheads of scan and handoff procedures are reduced by involving sensing methods that are capable of identifying PU transmission and maximum waiting time, $T_{MaxWait}$, factor before leaving a busy channel. In addition, using more highly accurate methods in QACR-MAC improves protection for PUs. The QACR-MAC parameters can be adjusted to suit a varied range of operational requirements, and have been designed to comply with various performance mechanisms used in IEEE 802.11-based networks, such as data frame aggregation or fragmentation. QACR-MAC does not require a dedicated control channel or centralised coordination, making it suitable for an extensive range of applications.

6.5. Simulation results

The simulation part of this thesis, in Chapter 5, has two core objectives: to study QoS degradation caused by sensing in CR networks based on IEEE 802.11, and to evaluate proposed solutions. During this study, none of the available simulation tools come ready with the support of a CR network based on IEEE 802.11af. In this work, Riverbed Modeler was used for implementing a CR node based on CSMA/CA with enhanced distributed channel access (EDCA) to employ different possible sensing parameters and strategies. The codes and design used to implement the CR node are documented in Appendix B and can be used in any future studies. The simulation results prove that fixed sensing strategies are not suitable when requiring high accuracy because they cause significant QoS degradation in the running application, in particular, voice applications that are sensitive to delay. In other words, the IEEE 802.11e (EDCA) QoS mechanism gradually loses its efficiency as the sensing duration is increased. In contrast, when a sensing method characterised by sensing duration and accuracy is selected according to QoS requirements, better performance is achieved. In particular, when the selection scheme was deployed as proposed in the QACR-MAC strategy, remarkable improvements in QoS were measured even when sensing methods of durations up to 300 ms were used. Moreover, the results from a range of simulation scenarios show that QACR-MAC performed well under diverse MAC and network settings. It offers a novel solution that integrates with IEEE 802.11e mechanisms to achieve better QoS and RFS utilisation in White-Fi networks relying on spectrum sensing.

6.6. Limitations and future work

The solutions provided in this thesis were developed under rather difficult circumstances as no extra control channel was available, nor a GDB or centralised controller. While the solutions can be deployed in a wide range of networks, the results reported in this thesis represent their minimum bounds. Wider and better performance may be achieved when features unavailable for this study are used. For example, using aggregated channels and new antenna technologies, such as multi-input multi-output (MIMO) may lead to higher throughput. The current Riverbed simulation tool does not fully support these features, which poses a hindrance to testing them. Future work should aim to expand the proposed fuzzy logic decision-making mechanism to consider such features in one of the existing input parameters, or to investigate whether their use will possibly require more inputs and outputs. Another possible improvement could be achieved by the using cooperative sensing approach in QACR-MAC. Usually, this is based on the cooperative SUs employing the same sensing technique, but in QACR-MAC SUs may use different techniques using high-accuracy sensing. This will lead to more accurate and efficient use of SU resources, beyond the typical purpose of addressing the problem of a hidden PU. However, how to minimise the overhead involved in exchanging local sensing outcomes requires more study in the future.

References

[1] (2013, 13- Apr- 2018). *Australian radiofrequency spectrum allocation chart*. Available: http://www.acma.gov.au/~/media/Spectrum%20Transformation%20and%20Government/Publication/pdf/spectrum_chart2013%20pdf.pdf

[2] L. Cui and M. B. Weiss, "Can unlicensed bands be used by unlicensed usage," *Paper for the 41st annual TPRC, September,* pp. 27-29, 2013.

[3] Y. SiXing, C. Dawei, Z. Qian, L. Mingyan, and L. ShuFang, "Mining Spectrum Usage Data: A Large-Scale Spectrum Measurement Study," *Mobile Computing, IEEE Transactions on,* vol. 11, pp. 1033-1046, 2012.

[4] A. Palaios, J. Riihijarvi, P. Mahonen, V. Atanasovski, L. Gavrilovska, P. Van Wesemael*, et al.*, "Two days of European spectrum: Preliminary analysis of concurrent spectrum use in seven European sites in GSM and ISM bands," in *Communications (ICC), 2013 IEEE International Conference on*, Budapest, Hungary, 2013, pp. 2666-2671.

[5] M. H. Islam, C. L. Koh, S. W. Oh, X. Qing, Y. Y. Lai, C. Wang*, et al.*, "Spectrum survey in Singapore: Occupancy measurements and analyses," in *Cognitive Radio Oriented Wireless Networks and Communications, 2008. CrownCom 2008. 3rd International Conference on*, 2008, pp. 1-7.

[6] R. I. Chiang, G. B. Rowe, and K. W. Sowerby, "A quantitative analysis of spectral occupancy measurements for cognitive radio," in *Vehicular Technology Conference, 2007. VTC2007-Spring. IEEE 65th*, 2007, pp. 3016-3020.

[7] S. D. Barnes, P. A. Jansen van Vuuren, and B. T. Maharaj, "Spectrum occupancy investigation: Measurements in South Africa," *Measurement,* vol. 46, pp. 3098-3112, 11/ 2013.

[8] M. A. McHenry, P. A. Tenhula, D. McCloskey, D. A. Roberson, and C. S. Hood, "Chicago spectrum occupancy measurements & analysis and a long-term studies proposal," in *Proceedings of the first international workshop on Technology and policy for accessing spectrum*, 2006, p. 1.

[9] A. Al-Hourani, V. Trajković, S. Chandrasekharan, and S. Kandeepan, "Spectrum occupancy measurements for different urban environments," in *Networks and Communications (EuCNC), 2015 European Conference on*, 2015, pp. 97-102.

[10] (2011, 2/4/2018). *Towards 2020—Future spectrum requirements for mobile broadband*. Available: http://www.acma.gov.au/webwr/_assets/main/lib312084/ifc13_2011_toward_2020-future_spectrum_requirements.pdf

[11] M. T. Masonta, M. Mzyece, and N. Ntlatlapa, "Spectrum Decision in Cognitive Radio Networks: A Survey," *Communications Surveys & Tutorials, IEEE,* vol. 15, pp. 1088-1107, 2013.

[12] J. Mitola and G. Q. Maguire Jr, "Cognitive radio: making software radios more personal," *Personal Communications, IEEE,* vol. 6, pp. 13-18, 1999.

[13] FCC, "FCC, ET Docket No 03-222 Notice of proposed rule making and order," F. C. C. (FCC), Ed., ed. Washington, DC.: Tech. Rep., December, 2003.

[14] N. A. Lala, M. Uddin, and N. Sheikh, "Identification and Integration of QoS parameters in Cognitive Radio Networks using Fuzzy Logic," *Int. J. Emerg. Sci,* vol. 3, pp. 279-288, 2013.

[15] P. Steenkiste, D. Sicker, G. Minden, and D. Raychaudhuri, "Future directions in cognitive radio network research," Arlington, Virginia, USA2009.

[16] S. Haykin, "Cognitive radio: brain-empowered wireless communications," *Selected Areas in Communications, IEEE Journal on,* vol. 23, pp. 201-220, 2005.

[17] J. Mitola, "The software radio architecture," *Communications Magazine, IEEE,* vol. 33, pp. 26-38, 1995.

[18] S. Mao, Y. Huang, Y. Li, P. Agrawal, and J. Tugnait, "Introducing software defined radio into undergraduate wireless engineering curriculum through a hands-on approach," in *Proc. The 2013 ASEE Annual Conference*, Atlanta, Georgia, 2013, pp. 1-10.

[19] A. Kumar, J. Wang, K. Challapali, and K. Shin, "Analysis of QoS Provisioning in Cognitive Radio Networks: A Case Study," *Wireless Personal Communications,* vol. 57, pp. 53-71, 2011/03/01 2011.

[20] Z. Qing and B. M. Sadler, "A Survey of Dynamic Spectrum Access," *Signal Processing Magazine, IEEE,* vol. 24, pp. 79-89, 2007.

[21] I. F. Akyildiz, L. Won-Yeol, M. C. Vuran, and S. Mohanty, "A survey on spectrum management in cognitive radio networks," *Communications Magazine, IEEE,* vol. 46, pp. 40-48, 2008.

[22] J. Marinho and E. Monteiro, "Cognitive radio: survey on communication protocols, spectrum decision issues, and future research directions," *Wireless networks,* vol. 18, pp. 147-164, 2012/02/01 2012.

[23] U. Mir and Z. A. Bhatti, "Time triggered handoff schemes in cognitive radio networks: A survey," *Journal of Network and Computer Applications,* 2017/11/15/ 2017.

[24] Y. Bo, G. Feng, S. Yanyan, L. Chengnian, and G. Xinping, "Channel-Aware Access for Cognitive Radio Networks," *Vehicular Technology, IEEE Transactions on,* vol. 58, pp. 3726-3737, 2009.

[25] E. Biglieri and M. Lops, "Multiuser detection in a dynamic environment–Part I: User identification and data detection," *Information Theory, IEEE Transactions on,* vol. 53, pp. 3158-3170, 2007.

[26] B. Zhao, Y. Chen, C. He, and L. Jiang, "Performance analysis of spectrum sensing with multiple primary users," *IEEE Transactions on Vehicular Technology,* vol. 61, pp. 914-918, 2012.

[27] S. Chaudhari, "Spectrum sensing for cognitive radios: Algorithms, performance, and limitations," DOCTORAL DISSERTATIONS, Department of Signal Processing and Acoustics, Aalto University, FINLAND, 2012.

[28] Y. Zeng, Y.-C. Liang, A. T. Hoang, and R. Zhang, "A review on spectrum sensing for cognitive radio: challenges and solutions," *EURASIP J. Adv. Signal Process,* vol. 2010, pp. 2-2, 2010.

[29] A. F. Molisch, L. J. Greenstein, and M. Shafi, "Propagation issues for cognitive radio," *Proceedings of the IEEE,* vol. 97, pp. 787-804, 2009.

[30] T. Yucek and H. Arslan, "A survey of spectrum sensing algorithms for cognitive radio applications," *Communications Surveys & Tutorials, IEEE,* vol. 11, pp. 116-130, 2009.

[31] H. Sun, A. Nallanathan, C.-X. Wang, and Y. Chen, "Wideband spectrum sensing for cognitive radio networks: a survey," *Wireless Communications, IEEE,* vol. 20, pp. 74-81, 2013.

[32] C. Kae Won, "Adaptive Sensing Technique to Maximize Spectrum Utilization in Cognitive Radio," *Vehicular Technology, IEEE Transactions on,* vol. 59, pp. 992-998, 2010.

[33] K. Cichoń, A. Kliks, and H. Bogucka, "Energy-Efficient Cooperative Spectrum Sensing: A Survey," *IEEE Communications Surveys & Tutorials,* vol. 18, pp. 1861-1886, 2016.

[34] D. Gurney, G. Buchwald, L. Ecklund, S. Kuffner, and J. Grosspietsch, "Geo-location database techniques for incumbent protection in the TV white space," in *New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008. 3rd IEEE Symposium on*, 2008, pp. 1-9.

[35] R. Murty, R. Chandra, T. Moscibroda, and P. Bahl, "SenseLess: A Database-Driven White Spaces Network," *Mobile Computing, IEEE Transactions on,* vol. 11, pp. 189-203, 2012.

[36] M. Nekovee, "Cognitive Radio Access to TV White Spaces: Spectrum Opportunities, Commercial Applications and Remaining Technology Challenges," in *New Frontiers in Dynamic Spectrum, 2010 IEEE Symposium on*, Singapore, 2010, pp. 1-10.

[37] S. Haddadi, H. Saeedi, and K. Navaie, "Channel coding adoption versus increasing sensing time in secondary service to manage the effect of imperfect spectrum sensing in cognitive radio networks," in *Communication and Information Theory (IWCIT), 2013 Iran Workshop on*, 2013, pp. 1-5.

[38] M. Höyhtyä, S. Pollin, and A. Mämmelä, "Improving the performance of cognitive radios through classification, learning, and predictive channel selection," *Advances in Electronics and Telecommunications,* vol. 2, pp. 28-38, 2011.

[39] M. Bkassiny, L. Yang, and S. K. Jayaweera, "A Survey on Machine-Learning Techniques in Cognitive Radios," *Communications Surveys & Tutorials, IEEE,* vol. 15, pp. 1136-1159, 2013.

[40] J. Mitola, "Cognitive Radio---An Integrated Agent Architecture for Software Defined Radio," Doctoral, 2000.

[41] S. Jayaweera and C. Christodoulou, "Radiobots: Architecture, algorithms and realtime reconfigurable antenna designs for autonomous, self-learning future cognitive radios," 2011.

[42] L. Gavrilovska, V. Atanasovski, I. Macaluso, and L. A. DaSilva, "Learning and Reasoning in Cognitive Radio Networks," *Communications Surveys & Tutorials, IEEE,* vol. 15, pp. 1761-1777, 2013.

[43] X. Liu, R. Zhu, B. Jalaian, and Y. Sun, "Dynamic Spectrum Access Algorithm Based on Game Theory in Cognitive Radio Networks," *Mobile Networks and Applications,* vol. 20, pp. 817-827, December 01 2015.

[44] H. Yuqing, W. Jiao, and J. Hong, "Modeling of Learning Inference and Decision-Making Engine in Cognitive Radio," in *Networks Security Wireless Communications and Trusted Computing (NSWCTC), 2010 Second International Conference on*, 2010, pp. 258-261.

[45] V. K. Tumuluru, P. Wang, and D. Niyato, "A neural network based spectrum prediction scheme for cognitive radio," in *Communications (ICC), 2010 IEEE International Conference on*, 2010, pp. 1-5.

[46] Z. Zhongshan, L. Keping, and W. Jianping, "Self-organization paradigms and optimization approaches for cognitive radio technologies: a survey," *Wireless Communications, IEEE,* vol. 20, pp. 36-42, 2013.

[47] X. Yan, Y. Ahmet Şekercioğlu, and S. Narayanan, "A survey of vertical handover decision algorithms in Fourth Generation heterogeneous wireless networks," *Computer Networks,* vol. 54, pp. 1848-1863, 2010.

[48] W.-Y. Lee and I. F. Akyildiz, "Spectrum-aware mobility management in cognitive radio cellular networks," *Mobile Computing, IEEE Transactions on,* vol. 11, pp. 529-542, 2012.

[49] O. A. Mohammed, K. El-Khatib, and M. V. Martin, "A Survey of Cognitive Radio Management Functions," in *COCORA 2014, The Fourth International Conference on Advances in Cognitive Radio*, Nice, France, 2014, pp. 6-13.

[50] S. Pandit and G. Singh, "An overview of spectrum sharing techniques in cognitive radio communication system," *Wireless Networks,* vol. 23, pp. 497-518, February 01 2017.

[51] W. Yuan and D. H. K. Tsang, "Distributed Power Allocation Algorithm for Spectrum Sharing Cognitive Radio Networks with QoS Guarantee," in *INFOCOM 2009, IEEE*, de Janeiro, Brazil, 2009, pp. 981-989.

[52] I. F. Akyildiz, W.-Y. Lee, and K. R. Chowdhury, "CRAHNs: Cognitive radio ad hoc networks," *Ad Hoc Networks,* vol. 7, pp. 810-836, 2009.

[53] M. Cesana, F. Cuomo, and E. Ekici, "Routing in cognitive radio networks: Challenges and solutions," *Ad Hoc Networks,* vol. 9, pp. 228-248, 2011.

[54] S. Shariat, "Spectrum-aware routing in cognitive radio MANETs," University of Surrey, 2018.

[55] L. Changqing, F. R. Yu, J. Hong, and V. C. M. Leung, "Cross-Layer Design for TCP Performance Improvement in Cognitive Radio Networks," *Vehicular Technology, IEEE Transactions on,* vol. 59, pp. 2485-2495, 2010.

[56] P. Zhang, K. Shi, S. Lin, J. Wang, and G. Xu, "Study on TCP Performance Improvement in Dynamic Spectrum Access Networks Based on Spectrum Switch Estimation," 2015.

[57] G. Baldini, T. Sturman, A. R. Biswas, R. Leschhorn, G. Godor, and M. Street, "Security Aspects in Software Defined Radio and Cognitive Radio Networks: A Survey and A Way Ahead," *Communications Surveys & Tutorials, IEEE,* vol. 14, pp. 355-379, 2012.

[58] J. Marinho, J. Granjal, and E. Monteiro, "A survey on security attacks and countermeasures with primary user detection in cognitive radio networks," *EURASIP Journal on Information Security,* vol. 2015, pp. 1-14, 2015.

[59] K. Arshad, R. Mackenzie, U. Celentano, A. Drozdy, S. Leveil, G. Mange*, et al.*, "Resource management for QoS support in cognitive radio networks," *IEEE Communications Magazine,* vol. 52, pp. 114-120, 2014.

[60] A. Sultana, X. Fernando, and L. Zhao, "An overview of medium access control strategies for opportunistic spectrum access in cognitive radio networks," *Peer-to-Peer Networking and Applications,* pp. 1-29, 2016.

[61] F. C. Commission, "Second Memorandum opinion and Order (FCC 10-174)," *US Govt. Printing Office, Washington, DC,* vol. 23, 2010.

[62] "IEEE Draft Standard for Information technology Telecommunications and information exchange between systems Local and metropolitan area networks Specific requirements Part 22.1: Standard to Enhance Harmful Interference Protection for Low Power Licensed Devices Operating in TV Broadcast Bands," *IEEE P802.22.1/D7, May 2010,* pp. 1-0, 2010.

[63] "ISO/IEC/IEEE International Standard - Information technology -- Telecommunications and information exchange between systems -- Local and metropolitan area networks -- Specific requirements -- Part 22: Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Policies and Procedures for Operation in the TV Bands AMENDMENT 2: Enhancement for broadband services and monitoring applications," *ISO/IEC/IEEE 8802-22.2:2015/Amd.2:2017(E),* pp. 1-302, 2018.

[64] IEEE, "ISO/IEC/IEEE International Standard - Information technology -- Telecommunications

and information exchange between systems -- Local and metropolitan area networks -- Specific requirements -- Part 22: Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications: Policies and Procedures for Operation in the TV Bands AMENDMENT 2: Enhancement for broadband services and monitoring applications," in *ISO/IEC/IEEE 8802-22.2:2015/Amd.2:2017(E)*, ed, 2018, pp. 1-302.

[65] IEEE, "IEEE Standard for Information technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment 5: Television White Spaces (TVWS) Operation," *IEEE Std 802.11af-2013 (Amendment to IEEE Std 802.11-2012, as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11aa-2012, IEEE Std 802.11ad-2012, and IEEE Std 802.11ac-2013),* pp. 1-198, 2014.

[66] A. Khattab and M. A. Bayoumi, "An overview of IEEE standardization efforts for cognitive radio networks," in *2015 IEEE International Symposium on Circuits and Systems (ISCAS)*, 2015, pp. 982-985.

[67] Y. Han, E. Ekici, H. Kremo, and O. Altintas, "A survey of MAC issues for TV white space access," *Ad Hoc Networks,* vol. 27, pp. 195-218, 2015.

[68] I. D. S. C. (DySPAN-SC). (2018, 25 April 2018). *IEEE Standards Association*. Available: http://grouper.ieee.org/groups/dyspan/

[69] I. Kakalou, K. E. Psannis, P. Krawiec, and R. Badea, "Cognitive Radio Network and Network Service Chaining toward 5G: Challenges and Requirements," *IEEE Communications Magazine,* vol. 55, pp. 145-151, 2017.

[70] R. Zhang, M. Wang, L. X. Cai, Z. Zheng, and X. Shen, "LTE-unlicensed: the future of spectrum aggregation for cellular networks," *Wireless Communications, IEEE,* vol. 22, pp. 150-159, 2015.

[71] P. Willemen, D. Laselva, Y. Wang, I. Kovács, R. Djapic, and I. Moerman, "SON for LTE-WLAN access network selection: design and performance," *EURASIP Journal on Wireless Communications and Networking,* vol. 2016, pp. 1-16, 2016.

[72] H. Kim and K. G. Shin, "In-band spectrum sensing in cognitive radio networks: energy detection or feature detection?," in *Proceedings of the 14th ACM international conference on Mobile computing and networking*, 2008, pp. 14-25.

[73] R. Tandra and A. Sahai, "SNR Walls for Signal Detection," *Selected Topics in Signal Processing, IEEE Journal of,* vol. 2, pp. 4-17, 2008.

[74] D. Cabric, S. M. Mishra, and R. W. Brodersen, "Implementation issues in spectrum sensing for cognitive radios," in *Signals, Systems and Computers, 2004. Conference Record of the Thirty-Eighth Asilomar Conference on*, 2004, pp. 772-776 Vol.1.

[75] Y. Zeng and Y.-C. Liang, "Spectrum-sensing algorithms for cognitive radio based on statistical covariances," *Vehicular Technology, IEEE Transactions on,* vol. 58, pp. 1804-1815, 2009.

[76] D. B. Rawat and G. Yan, "Spectrum sensing methods and dynamic spectrum sharing in cognitive radio networks: A survey," *International Journal of Research and Reviews in Wireless Sensor Networks,* vol. 1, pp. 1-13, 2011.

[77] J. Lundén, V. Koivunen, A. Huttunen, and H. V. Poor, "Spectrum sensing in cognitive radios based on multiple cyclic frequencies," in *Cognitive Radio Oriented Wireless Networks and Communications, 2007. CrownCom 2007. 2nd International Conference on*, 2007, pp. 37-43.

[78] S. Enserink and D. Cochran, "A cyclostationary feature detector," in *Signals, Systems and*

Computers, 1994. 1994 Conference Record of the Twenty-Eighth Asilomar Conference on, 1994, pp. 806-810.

[79] Z. Yonghong and L. Ying-Chang, "Robustness of the cyclostationary detection to cyclic frequency mismatch," in *Personal Indoor and Mobile Radio Communications (PIMRC), 2010 IEEE 21st International Symposium on*, 2010, pp. 2704-2709.

[80] M. Bkassiny, S. K. Jayaweera, L. Yang, and K. A. Avery, "Blind cyclostationary feature detection based spectrum sensing for autonomous self-learning cognitive radios," in *Communications (ICC), 2012 IEEE International Conference on*, 2012, pp. 1507-1511.

[81] A. Nafkha, B. Aziz, M. Naoues, and A. Kliks, "Cyclostationarity-based versus eigenvalues-based algorithms for spectrum sensing in cognitive radio systems: Experimental evaluation using GNU radio and USRP," in *Wireless and Mobile Computing, Networking and Communications (WiMob), 2015 IEEE 11th International Conference on*, 2015, pp. 310-315.

[82] M. Yang, Y. Li, X. Liu, and W. Tang, "Cyclostationary feature detection based spectrum sensing algorithm under complicated electromagnetic environment in cognitive radio networks," *China Communications,* vol. 12, pp. 35-44, 2015.

[83] H. Tang, "Some physical layer issues of wide-band cognitive radio systems," in *New frontiers in dynamic spectrum access networks, 2005. DySPAN 2005. 2005 first IEEE international symposium on*, 2005, pp. 151-159.

[84] R. K. Sharma and J. W. Wallace, "Correlation-based sensing for cognitive radio networks: Bounds and experimental assessment," *IEEE Sensors Journal,* vol. 11, pp. 657-666, 2011.

[85] T. Yucek and H. Arslan, "Spectrum characterization for opportunistic cognitive radio systems," in *Military Communications Conference, 2006. MILCOM 2006. IEEE*, 2006, pp. 1-6.

[86] T. Farnham, G. Clemo, R. Haines, E. Seidel, A. Benamar, S. Billington*, et al.*, "Ist-trust: A perspective on the reconfiguration of future mobile terminals using software download," in *Personal, Indoor and Mobile Radio Communications, 2000. PIMRC 2000. The 11th IEEE International Symposium on*, 2000, pp. 1054-1059.

[87] S. Shobana, R. Saravanan, and R. Muthaiah, "Optimal Spectrum Sensing Approach on Cognitive Radio Systems," 2013.

[88] S. Shobana, R. Saravanan, and R. Muthaiah, "Matched Filter Based Spectrum Sensing on Cognitive Radio for OFDM WLANs," *International Journal of Engineering and Technology (IJET),* vol. 5, 2013.

[89] W. Chien-Min, S. Hui-Kai, L. Maw-Lin, L. Yi-Ching, and L. Chih-Pin, "Cooperative Power and Contention Control MAC Protocol in Multichannel Cognitive Radio Ad Hoc Networks," in *Innovative Mobile and Internet Services in Ubiquitous Computing (IMIS), 2014 Eighth International Conference on*, 2014, pp. 305-309.

[90] B. Wang, Z. Bai, Y. Xu, P. Dong, and K. Kwak, "Dynamical clustering cooperative spectrum sensing with bandwidth constraints in CR systems," in *Advanced Communication Technology (ICACT), 2014 16th International Conference on*, Pyeongchang, South Korea, 2014, pp. 244-248.

[91] G. Sharma and R. Sharma, "Distributed cooperative spectrum sensing over different fading channels in cognitive radio," in *2017 International Conference on Computer, Communications and Electronics (Comptelix)*, 2017, pp. 107-111.

[92] X. Dongyue, E. Ekici, and M. C. Vuran, "Cooperative Spectrum Sensing in Cognitive Radio Networks Using Multidimensional Correlations," *Wireless Communications, IEEE Transactions*

on, vol. 13, pp. 1832-1843, 2014.

[93] I. F. Akyildiz, B. F. Lo, and R. Balakrishnan, "Cooperative spectrum sensing in cognitive radio networks: A survey," *Physical Communication,* vol. 4, pp. 40-62, 2011.

[94] Z. Yulong, Y. Yu-Dong, and Z. Baoyu, "Cooperative relay techniques for cognitive radio systems: Spectrum sensing and secondary user transmissions," *Communications Magazine, IEEE,* vol. 50, pp. 98-103, 2012.

[95] M. Subhedar and G. Birajdar, "Spectrum sensing techniques in cognitive radio networks: A survey," *International Journal of Next-Generation Networks,* vol. 3, pp. 37-51, 2011.

[96] J. L. Mauri, K. Z. Ghafoor, D. B. Rawat, and J. M. A. Perez, *Cognitive Networks: Applications and Deployments*: CRC Press, 2014.

[97] T. MathWorks, "MATLAB," 2014a ed: The MathWorks, Inc., 2014.

[98] S. Kalamkar, "Energy Detection Simulation: Rayleigh Channel," version 1.0 ed: The MathWorks, Inc., 2015, p. This code simulates ROC curve for the energy detection in Rayleigh channel.

[99] S. Sesham and S. Sabat, "Spectrum Sensing for Cognitive Radio Networks," in *White Space Communication*, A. K. Mishra and D. Lloyd Johnson, Eds., ed: Springer International Publishing, 2015, pp. 117-151.

[100] W. Lee, K. Srisomboon, and A. Prayote, "Fast Spectrum Sensing with Coordinate System in Cognitive Radio Networks," *ETRI Journal,* vol. 37, pp. 491-501, 2015.

[101] S. S. Kalamkar, A. Banerjee, and A. K. Gupta, "SNR wall for generalized energy detection under noise uncertainty in cognitive radio," in *2013 19th Asia-Pacific Conference on Communications (APCC)*, 2013, pp. 375-380.

[102] R. Tandra and A. Sahai, "SNR Walls for Feature Detectors," in *New Frontiers in Dynamic Spectrum Access Networks, 2007. DySPAN 2007. 2nd IEEE International Symposium on*, 2007, pp. 559-570.

[103] M. Kyryk, L. Matiishyn, V. Yanyshyn, and V. Havronskyy, "Performance comparison of cognitive radio networks spectrum sensing methods," in *2016 13th International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science (TCSET)*, 2016, pp. 597-600.

[104] A. Gurijala and C. Molina, "Defining and monitoring QOS metrics in the next generation wireless networks," in *Telecommunications Quality of Services: The Business of Success, 2004. QoS 2004. IEE*, 2004, pp. 37-42.

[105] Y. Chen, T. Farley, and N. Ye, "QoS requirements of network applications on the Internet," *Information, Knowledge, Systems Management,* vol. 4, pp. 55-76, 2004.

[106] F. Farid, S. Shahrestani, and R. Chun, "Application Based Handover: An Energy-Efficient Approach for Heterogeneous Networks," in *Computational Science and Engineering (CSE), 2013 IEEE 16th International Conference on*, 2013, pp. 1185-1190.

[107] P. Yiyang, L. Ying-Chang, K. C. Teh, and L. Kwok Hung, "How much time is needed for wideband spectrum sensing?," *Wireless Communications, IEEE Transactions on,* vol. 8, pp. 5466-5471, 2009.

[108] H. Xin-Lin, W. Gang, H. Fei, and S. Kumar, "The Impact of Spectrum Sensing Frequency and Packet-Loading Scheme on Multimedia Transmission Over Cognitive Radio Networks,"

Multimedia, IEEE Transactions on, vol. 13, pp. 748-761, 2011.

[109] A. Ghasemi and E. S. Sousa, "Spectrum sensing in cognitive radio networks: requirements, challenges and design trade-offs," *Communications Magazine, IEEE,* vol. 46, pp. 32-39, 2008.

[110] L. Ying-Chang, Z. Yonghong, E. C. Y. Peh, and H. Anh Tuan, "Sensing-Throughput Tradeoff for Cognitive Radio Networks," *Wireless Communications, IEEE Transactions on,* vol. 7, pp. 1326-1337, 2008.

[111] P. Yiyang, H. Anh Tuan, and L. Ying-Chang, "Sensing-Throughput Tradeoff in Cognitive Radio Networks: How Frequently Should Spectrum Sensing be Carried Out?," in *Personal, Indoor and Mobile Radio Communications, 2007. PIMRC 2007. IEEE 18th International Symposium on*, 2007, pp. 1-5.

[112] K. Hamdi and K. B. Letaief, "Power, Sensing Time, and Throughput Tradeoffs in Cognitive Radio Systems: A Cross-Layer Approach," in *2009 IEEE Wireless Communications and Networking Conference*, 2009, pp. 1-5.

[113] G. Verma and O. P. Sahu, "Removal of Sensing-Throughput Tradeoff Barrier in Cognitive Radio Networks," *Wireless Personal Communications,* pp. 1-14, 2016.

[114] A. El Shafie and A. Sultan, "Optimal selection of spectrum sensing duration for an energy harvesting cognitive radio," in *Global Communications Conference (GLOBECOM), 2013 IEEE*, 2013, pp. 1020-1025.

[115] G. Noh, J. Lee, H. Wang, S. Kim, S. Choi, and D. Hong, "Throughput Analysis and Optimization of Sensing-Based Cognitive Radio Systems With Markovian Traffic," *IEEE Transactions on Vehicular Technology,* vol. 59, pp. 4163-4169, 2010.

[116] H. Hu, H. Zhang, and H. Yu, "Delay QoS guaranteed cooperative spectrum sensing in cognitive radio networks," *AEU - International Journal of Electronics and Communications,* vol. 67, pp. 804-807, 2013/09/01/ 2013.

[117] S. Gezici, H. Celebi, H. V. Poor, and H. Arslan, "Fundamental limits on time delay estimation in dispersed spectrum cognitive radio systems," *IEEE Transactions on Wireless Communications,* vol. 8, pp. 78-83, 2009.

[118] W. Yin, P. Ren, Q. Du, and Y. Wang, "Delay and Throughput Oriented Continuous Spectrum Sensing Schemes in Cognitive Radio Networks," *IEEE Transactions on Wireless Communications,* vol. 11, pp. 2148-2159, 2012.

[119] M. E. Bayrakdar and A. Çalhan, "Delay characteristics of TDMA medium access control protocol for cognitive radio networks," in *2016 XIth International Scientific and Technical Conference Computer Sciences and Information Technologies (CSIT)*, 2016, pp. 66-69.

[120] A. Satheesh, S. H. Aswini, S. G. Lekshmi, S. Sagar, and M. Hareesh Kumar, "Spectrum sensing techniques A comparison between energy detector and cyclostationarity detector," in *2013 International Conference on Control Communication and Computing (ICCC)*, 2013, pp. 388-393.

[121] A. Bagwari and G. S. Tomar, "Enriched the Spectrum Sensing Performance of Estimated SNR Based Detector in Cognitive Radio Networks," 2015.

[122] D. Ariananda, M. Lakshmanan, and H. Nikookar, "A survey on spectrum sensing techniques for cognitive radio," in *Cognitive Radio and Advanced Spectrum Management, 2009. CogART 2009. Second International Workshop on*, 2009, pp. 74-79.

[123] S. L. Sabat, S. Srinu, A. Raveendranadh, and S. K. Udgata, *Spectrum sensing based on entropy*

estimation using cyclostationary features for cognitive radio. Bangalore, India: Paper presented at the fourth international conference on communication systems and networks (COMSNETS), 2012.

[124] R. Vadivelu, K. Sankaranarayanan, and V. Vijayakumari, "MATCHED FILTER BASED SPECTRUM SENSING FOR COGNITIVE RADIO AT LOW SIGNAL TO NOISE RATIO," *Journal of Theoretical & Applied Information Technology,* vol. 62, 2014.

[125] C. Cormio and K. R. Chowdhury, "A survey on MAC protocols for cognitive radio networks," *Ad Hoc Networks,* vol. 7, pp. 1315-1329, 2009.

[126] S. Y. Lien, C. C. Tseng, and K. C. Chen, "Carrier Sensing Based Multiple Access Protocols for Cognitive Radio Networks," in *2008 IEEE International Conference on Communications*, New York, 2008, pp. 3208-3214.

[127] IEEE, "IEEE Draft Standard for Information Technology -Telecommunications and information exchange between systems - Wireless Regional Area Networks (WRAN) - Specific requirements - Part 22: Cognitive Wireless RAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Policies and procedures for operation in the TV Bands," in *IEEE P802.22/D3.0, March 2011*, ed, 2011, pp. 1-664.

[128] C. Zou and C. Chigan, "A Game Theoretic DSA-Driven MAC Framework for Cognitive Radio Networks," in *2008 IEEE International Conference on Communications*, New York, 2008, pp. 4165-4169.

[129] L. Ma, C. C. Shen, and B. Ryu, "Single-Radio Adaptive Channel Algorithm for Spectrum Agile Wireless Ad Hoc Networks," in *2007 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, New York, 2007, pp. 547-558.

[130] J. Jia, Q. Zhang, and X. Shen, "HC-MAC: a hardware-constrained cognitive MAC for efficient spectrum management," *IEEE J Sel Areas Commun,* vol. 26, 2008.

[131] M. Liangping, H. Xiaofeng, and S. Chien-Chung, "Dynamic open spectrum sharing MAC protocol for wireless ad hoc networks," in *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005.*, 2005, pp. 203-213.

[132] P. Pawelczak, R. V. Prasad, X. Liang, and I. G. M. M. Niemegeers, "Cognitive radio emergency networks - requirements and design," in *First IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks, 2005. DySPAN 2005.*, 2005, pp. 601-606.

[133] C. Cordeiro and K. Challapali, "C-MAC: A Cognitive MAC Protocol for Multi-Channel Wireless Networks," in *2007 2nd IEEE International Symposium on New Frontiers in Dynamic Spectrum Access Networks*, 2007, pp. 147-157.

[134] B. Hamdaoui and K. G. Shin, "OS-MAC: an efficient MAC protocol for spectrum-agile wireless networks," *IEEE Trans Mob Comput,* vol. 7, 2008.

[135] Q. Zhao, L. Tong, A. Swami, and Y. Chen, "Decentralized cognitive MAC for opportunistic spectrum access in ad hoc networks: a POMDP framework," *IEEE J Sel Areas Commun,* vol. 25, 2007.

[136] Y. R. Kondareddy and P. Agrawal, "Synchronized MAC Protocol For Multi-Hop Cognitive Radio Networks," in *2008 IEEE International Conference on Communications*, 2008, pp. 3198-3202.

[137] H. Su and X. Zhang, "Opportunistic MAC Protocols for Cognitive Radio Based Wireless Networks," in *2007 41st Annual Conference on Information Sciences and Systems*, 2007, pp. 363-368.

[138] P. Ren, Y. Wang, Q. Du, and J. Xu, "A survey on dynamic spectrum access protocols for distributed cognitive wireless networks," *EURASIP J. Wireless Comm. and Networking,* vol. 2012, p. 60, 2012.

[139] IEEE, "IEEE Draft Standard for Information technology--Telecommunications and information exchange between systems Local and metropolitan area networks--Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," in *IEEE P802.11-REVmc/D5.0, January 2016 (Revision of IEEE Std 802.11-2012 as amended by IEEE Std 802.11ae-2012, IEEE Std 802.11aa-2012, IEEE Std 802.11ad-2012, IEEE Std 802.11ac-2013, and IEEE Std 802.11af-2013)*, ed, 2016, pp. 1-3766.

[140] Q. Chen, Y. C. Liang, M. Motani, and W. C. Wong, "CR-CSMA: A Random Access MAC Protocol for Cognitive Radio Networks," in *2009 IEEE 20th International Symposium on Personal, Indoor and Mobile Radio Communications*, 2009, pp. 486-490.

[141] Q. Chen, Y. C. Liang, M. Motani, and W. C. Wong, "A Two-Level MAC Protocol Strategy for Opportunistic Spectrum Access in Cognitive Radio Networks," *IEEE Transactions on Vehicular Technology,* vol. 60, pp. 2164-2180, 2011.

[142] L. X. Cai, Y. Liu, X. Shen, J. W. Mark, and D. Zhao, "Distributed QoS-Aware MAC for Multimedia over Cognitive Radio Networks," in *2010 IEEE Global Telecommunications Conference GLOBECOM 2010*, 2010, pp. 1-5.

[143] M. Luis, R. Oliveira, R. Dinis, and L. Bernardo, "A Novel Reservation-based MAC Scheme for Distributed Cognitive Radio Networks," *IEEE Transactions on Vehicular Technology,* vol. PP, pp. 1-1, 2016.

[144] T. Oo, N. Tran, D. Dang, and Z. Han, "OMF-MAC: an opportunistic matched filter-based MAC in cognitive radio networks," *IEEE Trans Vehicular Technol,* vol. 65, 2016.

[145] W. Ejaz, N. U. Hasan, S. Aslam, and H. S. Kim, "Fuzzy logic based spectrum sensing for cognitive radio networks," in *Next Generation Mobile Applications, Services and Technologies (NGMAST), 2011 5th International Conference on*, 2011, pp. 185-189.

[146] M. Matinmikko, J. Del Ser, T. Rauma, and M. Mustonen, "Fuzzy-Logic Based Framework for Spectrum Availability Assessment in Cognitive Radio Systems," *Selected Areas in Communications, IEEE Journal on,* vol. 31, pp. 2173-2184, 2013.

[147] M. Matinmikko, M. Mustonen, T. Rauma, and J. D. Ser, "Decision-making system for obtaining spectrum availability information in opportunistic networks," presented at the Proceedings of the 4th International Conference on Cognitive Radio and Advanced Spectrum Management, Barcelona, Spain, 2011.

[148] M. Luıs, R. Luıs, R. Oliveira, R. Dinis, and L. Bernardo, "On the performance of decentralized CR MAC protocols under heterogeneous channel sensing conditions," in *2015 IEEE Wireless Communications and Networking Conference Workshops (WCNCW)*, 2015, pp. 247-252.

[149] S. Atmaca, Ö. Şayli, J. Yuan, and A. Kavak, "Throughput Maximization of CSMA in Cognitive Radio Networks with Cooperative Spectrum Sensing," *Wireless Personal Communications,* vol. 92, pp. 1473-1492, February 01 2017.

[150] P. Gronsund, P. Pawelczak, J. Park, and D. Cabric, "System level performance of IEEE 802.22-2011 with sensing-based detection of wireless microphones," *IEEE Communications Magazine,* vol. 52, pp. 200-209, 2014.

[151] M. Matinmikko, T. Rauma, M. Mustonen, and J. Del Ser, "Architecture and approach for obtaining spectrum availability information," in *IEEE Vehicular Technology Conference*, 2011.

[152] B. Ahuja and G. Kaur, "A novel two stage improved spectrum sensing for cognitive radio systems," in *2016 3rd International Conference on Computing for Sustainable Global Development (INDIACom)*, 2016, pp. 1533-1538.

[153] M. Mushtaq, M. Khan, M. Naqvi, R. Khan, M. Khan, and O. F. Koudelka, "Cognitive Radios and Cognitive Networks: A short Introduction," 2013.

[154] S. RAJPUT, N. BOHRA, H. M. HARIS, P. KHALID, and S. Z. SHAH, "A Modified Energy Detection Based Spectrum Sensing Algorithm for Green Cognitive Radio Communication," *Mehran University Research Journal of Engineering & Technology,* vol. 34, 2015.

[155] IEEE, "ISO/IEC/IEEE International Standard -- Information technology -- Telecommunications and information exchange between systems -- Local and metropolitan area networks -- Specific requirements -- Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications AMENDMENT 5," *IS0/IEC/IEEE 8802-11:2012/Amd.5:2015(E) (Adoption of IEEE Std 802.11af-2014),* pp. 1-204, 2015.

[156] X. Kang, H. K. Garg, Y. C. Liang, and R. Zhang, "Optimal power allocation for OFDM-based cognitive radio with new primary transmission protection criteria," *IEEE Transactions on Wireless Communications,* vol. 9, pp. 2066-2075, 2010.

[157] L. Giupponi and A. I. Perez-Neira, "Fuzzy-based Spectrum Handoff in Cognitive Radio Networks," in *Cognitive Radio Oriented Wireless Networks and Communications, 2008. CrownCom 2008. 3rd International Conference on*, 2008, pp. 1-6.

[158] L. A. Zadeh, "Fuzzy sets," *Information and control,* vol. 8, pp. 338-353, 1965.

[159] E. H. Mamdani and S. Assilian, "An experiment in linguistic synthesis with a fuzzy logic controller," *International Journal of Man-Machine Studies,* vol. 7, pp. 1-13, 1975/01/01/ 1975.

[160] L. A. Zadeh, "Making computers think like people [fuzzy set theory]," *IEEE spectrum,* vol. 21, pp. 26-32, 1984.

[161] C. Kahraman, U. Kaymak, and A. Yazici, *Fuzzy Logic in Its 50th Year: new developments, directions and challenges* vol. 341: Springer, 2016.

[162] Y. Bai and D. Wang, "Fundamentals of fuzzy logic control—fuzzy sets, fuzzy rules and defuzzifications," in *Advanced Fuzzy Logic Technologies in Industrial Applications*, ed: Springer, 2006, pp. 17-36.

[163] S. Sivanandam, S. Sumathi, and S. Deepa, *Introduction to fuzzy logic using MATLAB* vol. 1: Springer, 2007.

[164] T. Takagi and M. Sugeno, "Fuzzy identification of systems and its applications to modeling and control," *IEEE transactions on systems, man, and cybernetics,* pp. 116-132, 1985.

[165] M. Sugeno and G. Kang, "Structure identification of fuzzy model," *Fuzzy sets and systems,* vol. 28, pp. 15-33, 1988.

[166] (2017, 11-NOV-2017). *Defuzzification Methods - MATLAB & Simulink Example - MathWorks United Kingdom*. Available: http://au.mathworks.com/help/fuzzy/examples/defuzzification-methods.html

[167] S. Roychowdhury and W. Pedrycz, "A survey of defuzzification strategies," *International Journal of intelligent systems,* vol. 16, pp. 679-695, 2001.

[168] IEEE, "IEEE Standard for Information Technology - Telecommunications and Information Exchange Between Systems - Local and Metropolitan Area Networks - Specific Requirements -

Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999),* pp. 1-1076, 2007.

[169] M. Jaiswal, A. K. Sharma, and V. Singh, "A Survey on Spectrum Sensing Techniques for Cognitive Radio," in *Proceedings of the Conference on Advances in Communication and Control Systems-2013*, 2013.

[170] C. Ghosh, S. Roy, and D. Cavalcanti, "Coexistence challenges for heterogeneous cognitive wireless networks in TV white spaces," *Wireless Communications, IEEE,* vol. 18, pp. 22-31, 2011.

[171] "IEEE Standard for Information technology--Telecommunications and information exchange between systems -- Local and metropolitan area networks -- Specific requirements -- Part 19: TV White Space Coexistence Methods," *IEEE Std 802.19.1-2014,* pp. 1-326, 2014.

[172] C. Cormio and K. R. Chowdhury, "Common control channel design for cognitive radio wireless ad hoc networks using adaptive frequency hopping," *Ad Hoc Networks,* vol. 8, pp. 430-438, 2010.

[173] P. Bahl, R. Chandra, T. Moscibroda, R. Murty, and M. Welsh, "White space networking with wi-fi like connectivity," *ACM SIGCOMM Computer Communication Review,* vol. 39, pp. 27-38, 2009.

[174] A. B. Flores, R. E. Guerra, E. W. Knightly, P. Ecclesine, and S. Pandey, "IEEE 802.11 af: a standard for TV white space spectrum sharing," *Communications Magazine, IEEE,* vol. 51, pp. 92-100, 2013.

[175] J. Li, T. Luo, X. Liu, and G. Yue, "Link maintenance scheme for OFDM-based cognitive radio Ad Hoc networks," *Communications, China,* vol. 12, pp. 42-51, 2015.

[176] J. Ansari, X. Zhang, P. M\, \#228, h\, \#246*, et al.*, "A decentralized MAC for opportunistic spectrum access in cognitive wireless networks," presented at the Proceedings of the 2010 ACM workshop on Cognitive radio networks, Chicago, Illinois, USA, 2010.

[177] Y. R. Kondareddy, P. Agrawal, and K. Sivalingam, "Cognitive radio network setup without a common control channel," in *Military Communications Conference, 2008. MILCOM 2008. IEEE*, 2008, pp. 1-6.

[178] J. W. Chong, D. K. Sung, and Y. Sung, "Cross-layer performance analysis for CSMA/CA protocols: impact of imperfect sensing," *Vehicular Technology, IEEE Transactions on,* vol. 59, pp. 1100-1108, 2010.

[179] F. Foukalas, G. T. Karetsos, and P. Chatzimisios, "Cross-layer Design of CSMA/CA with Spectrum Sensing for Cognitive Radio Networks," in *Wireless Communication Systems (ISWCS 2013), Proceedings of the Tenth International Symposium on*, 2013, pp. 1-5.

[180] S.-J. Kim, E. Dall'Anese, J. A. Bazerque, K. Rajawat, and G. B. Giannakis, "Advances in spectrum sensing and cross-layer design for cognitive radio networks," *E-Reference Signal Processing,* 2013.

[181] J.-L. Chen, S.-W. Liu, S.-L. Wu, and M.-C. Chen, "Cross-layer and cognitive QoS management system for next-generation networking," *International Journal of Communication Systems,* vol. 24, pp. 1150-1162, 2011.

[182] H. Saki, A. Shojaeifard, and M. Shikh-Bahaei, "Cross-layer resource allocation for video streaming over OFDMA cognitive radio networks with imperfect cross-link CSI," in *2014 International Conference on Computing, Networking and Communications, ICNC 2014*, Honolulu, HI, USA, 2014, pp. 98-104.

[183] M. Yazid, L. Bouallouche-Medjkoune, D. Aïssani, and L. Ziane-Khodja, "Analytical analysis of applying packet fragmentation mechanism on IEEE 802.11 b DCF network in non ideal channel with infinite load conditions," *Wireless networks,* vol. 20, pp. 917-934, 2014.

[184] A. Sidelnikov, J. Yu, and S. Choi, "Fragmentation/aggregation scheme for throughput enhancement of IEEE 802.11 n WLAN," in *The 3rd IEEE Asia Pacific Wireless Communications Symposium (APWCS 2006)*, Daejeon, South Korea, 2006.

[185] E. Charfi, L. Chaari, and L. Kamoun, "PHY/MAC Enhancements and QoS Mechanisms for Very High Throughput WLANs: A Survey," *Communications Surveys & Tutorials, IEEE,* vol. 15, pp. 1714-1735, 2013.

[186] B. Ginzburg and A. Kesselman, "Performance analysis of A-MPDU and A-MSDU aggregation in IEEE 802.11 n," in *Sarnoff symposium, 2007 IEEE*, 2007, pp. 1-5.

[187] O. Sharon and Y. Alpert, "MAC level Throughput comparison: 802.11 ac vs. 802.11 n," *Physical Communication,* vol. 12, pp. 33-49, 2014.

[188] M. Khan, T. A. Khan, and M. Beg, "RTS/CTS mechanism of MAC layer IEEE 802.11 WLAN in presence of hidden nodes," *International Journal of Engineering and Innovative Technology (IJEIT) Volume,* vol. 2, 2012.

[189] M. Yazid, L. Bouallouche-Medjkoune, D. Aïssani, N. Amrouche, and K. Bakli, "Analytical Analysis of Applying Packet Fragmentation Mechanism on Both Basic and RTS/CTS Access Methods of the IEEE 802.11b DCF Network Under Imperfect Channel and Finite Load Conditions," *Wireless Personal Communications,* vol. 77, pp. 477-506, 2014/07/01 2014.

[190] S. Shiann-Tsong, T. Chen, C. Jenhui, and Y. Fun, "The impact of RTS threshold on IEEE 802.11 MAC protocol," in *Parallel and Distributed Systems, 2002. Proceedings. Ninth International Conference on*, 2002, pp. 267-272.

[191] H. Alakoca, M. Karaca, and G. K. Kurt, "The impact of MAC layer parameters in system performance of 802.11ac multi-hop networks," in *Signal Processing and Communications Applications Conference (SIU), 2015 23th*, 2015, pp. 1441-1444.

[192] C. Chousidis, R. Nilavalan, and L. Lipan, "Expanding the use of CTS-to-Self mechanism for reliable broadcasting on IEEE 802.11 networks," in *Wireless Communications and Mobile Computing Conference (IWCMC), 2014 International*, 2014, pp. 1051-1056.

[193] A. Bhartia, M. Gowda, K. K. Chintalapudi, B. Radunovic, R. Ramjee, D. Chakrabarty*, et al.*, "WiFi-XL: Extending WiFi to Wide Areas in White Spaces," *Microsoft Research TechReport TR-2014-132,* 2014.

[194] H. A. B. Salameh, M. M. Krunz, and O. Younis, *MAC protocol for opportunistic cognitive radio networks with soft guarantees*. New York: IEEE, 2009.

[195] L. Chouhan and A. Trivedi, "Performance study of a CSMA based multiuser MAC protocol for cognitive radio networks: analysis of channel utilization and opportunity perspective," *Wireless Networks,* vol. 22, pp. 33-47, 2016.

[196] IEEE, "IEEE Standard for Information technology--Local and metropolitan area networks--Specific requirements--Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications - Amendment 8: Medium Access Control (MAC) Quality of Service Enhancements," *IEEE Std 802.11e-2005 (Amendment to IEEE Std 802.11, 1999 Edition (Reaff 2003),* pp. 1-212, 2005.

[197] L. Gabrielli and S. Squartini, "Wireless Networked Music Performance," in *Wireless Networked Music Performance*, ed: Springer, 2016, pp. 53-92.

[198] S. Mangold, S. Choi, G. R. Hiertz, O. Klein, and B. Walke, "Analysis of IEEE 802.11 e for QoS support in wireless LANs," *Wireless Communications, IEEE,* vol. 10, pp. 40-50, 2003.

[199] "IEEE Standard for Information technology--Telecommunications and information exchange between systems Local and metropolitan area networks--Specific requirements Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications," *IEEE Std 802.11-2012 (Revision of IEEE Std 802.11-2007),* pp. 1-2793, 2012.

[200] I. Inan, F. Keceli, and E. Ayanoglu, "Modeling the 802.11 e enhanced distributed channel access function," in *Global Telecommunications Conference, 2007. GLOBECOM'07. IEEE*, 2007, pp. 2546-2551.

[201] "ISO/IEC/IEEE International Standard -- Information technology -- Telecommunications and information exchange between systems -- Local and metropolitan area networks -- Specific requirements -- Part 11: Wireless LAN medium access control (MAC) and physical layer (PHY) specifications AMENDMENT 5," *IS0/IEC/IEEE 8802-11:2012/Amd.5:2015(E) (Adoption of IEEE Std 802.11af-2014),* pp. 1-204, 2015.

[202] X. Yang, "IEEE 802.11e: QoS provisioning at the MAC layer," *Wireless Communications, IEEE,* vol. 11, pp. 72-79, 2004.

[203] G. Bianchi, I. Tinnirello, and L. Scalia, "Understanding 802.11 e contention-based prioritization mechanisms and their coexistence with legacy 802.11 stations," *Network, IEEE,* vol. 19, pp. 28-34, 2005.

[204] A. BAGWARI, J. KANTI, and G. S. TOMAR, "Novel Spectrum detector for IEEE 802.22 Wireless Regional Area Network," *International Journal of Smart Device and Appliance,* vol. 3, 2015.

[205] "IEEE Standard for Spectrum Sensing Interfaces and Data Structures for Dynamic Spectrum Access and Other Advanced Radio Communication Systems - Amendment 1: Procedures, Protocols, and Data Archive Enhanced Interfaces," *IEEE Std 1900.6a-2014 (Amendment to IEEE Std 1900.6-2011),* pp. 1-52, 2014.

[206] M. Q. Khan and S. H. Andresen, "An Intelligent Scan Mechanism for 802.11 Networks by Using Media Independent Information Server (MIIS)," in *2011 IEEE Workshops of International Conference on Advanced Information Networking and Applications*, 2011, pp. 221-225.

[207] B. Bawazeer, N. Akkari, G. Aldabbagh, and N. Dimitriou, "Reducing scanning delay for WiFi-to-WhiteFi handovers," in *2015 Fifth International Conference on Digital Information and Communication Technology and its Applications (DICTAP)*, 2015, pp. 49-53.

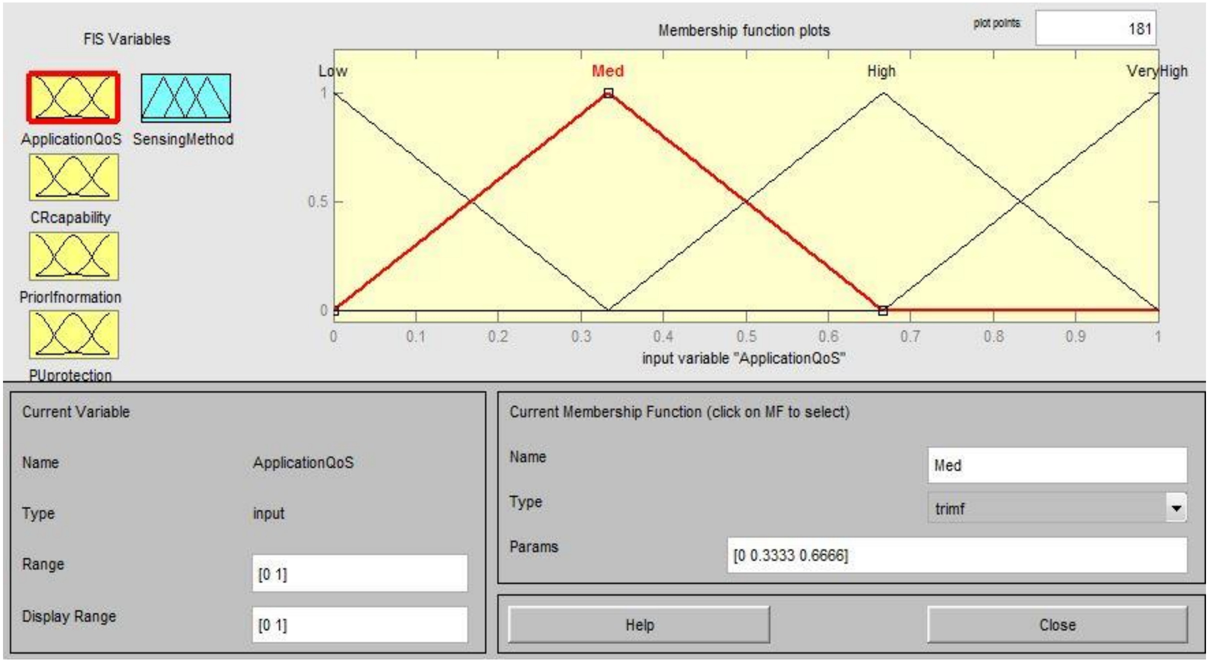
[208] TETCOS. (2016, 13- Apr- 2018). *NetSim*. Available: http://www.tetcos.com/

[209] (2013, 23- Nov- 2017). *Riverbed Modeler Support*. Available: https://supportkb.riverbed.com/support/index?page=content&id=S21031&actp=search&viewlocale=en_US&searchid=1513312172092

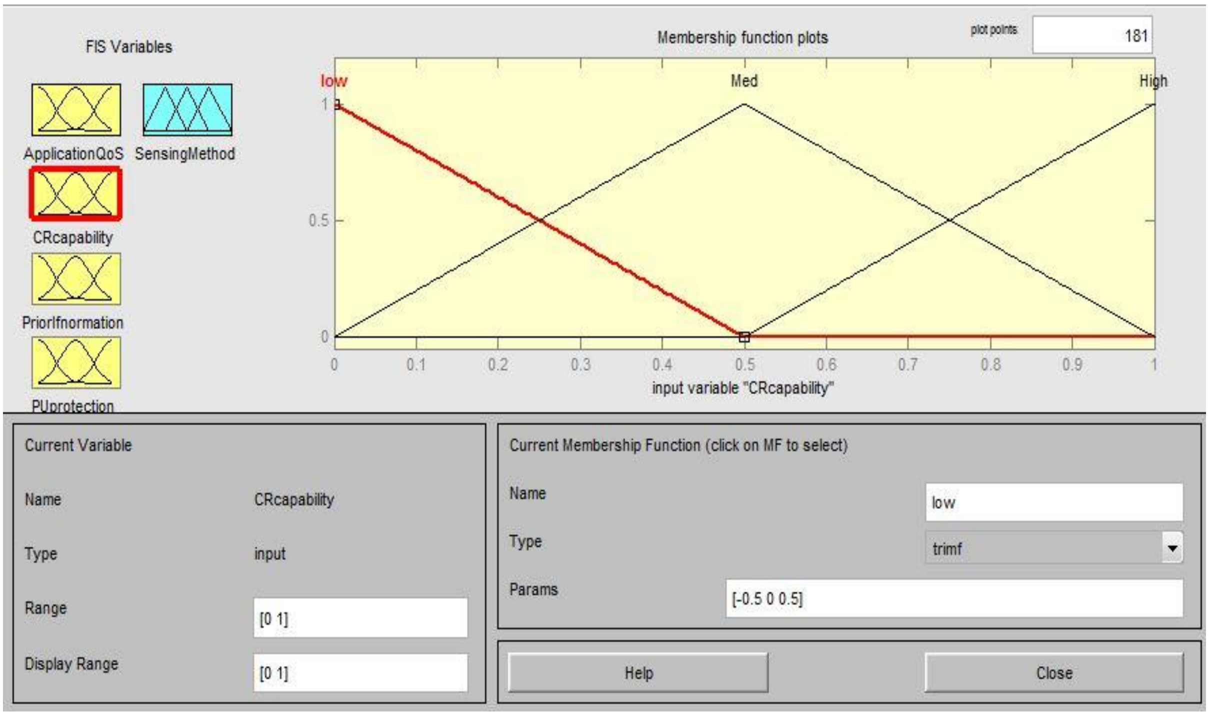
[210] N. Faruk, M. I. Gumel, A. A. Oloyode, and A. A. Ayeni, "Performance analysis of hybrid MAC protocol for cognitive radio networks," *International Journal of Communications, Network and System Sciences,* vol. 6, p. 18, 2013.

Appendix A: MATLAB Implementations

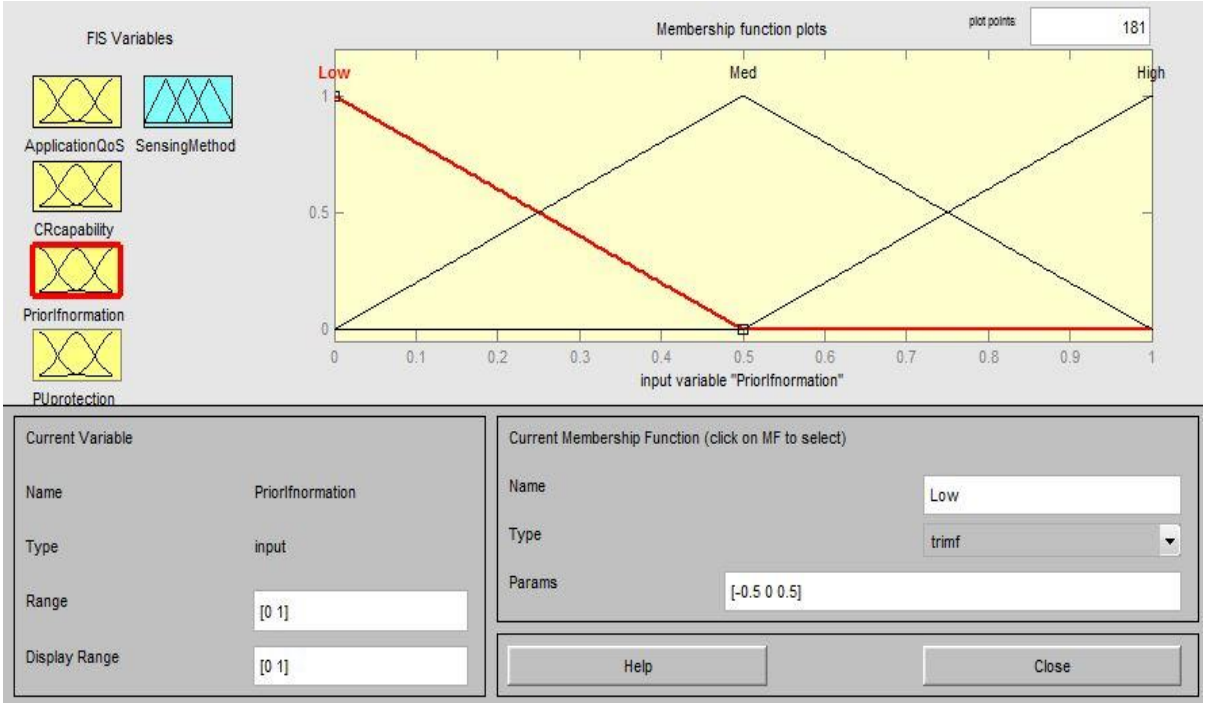
Appendix A 1 Input membership function for Application QoS



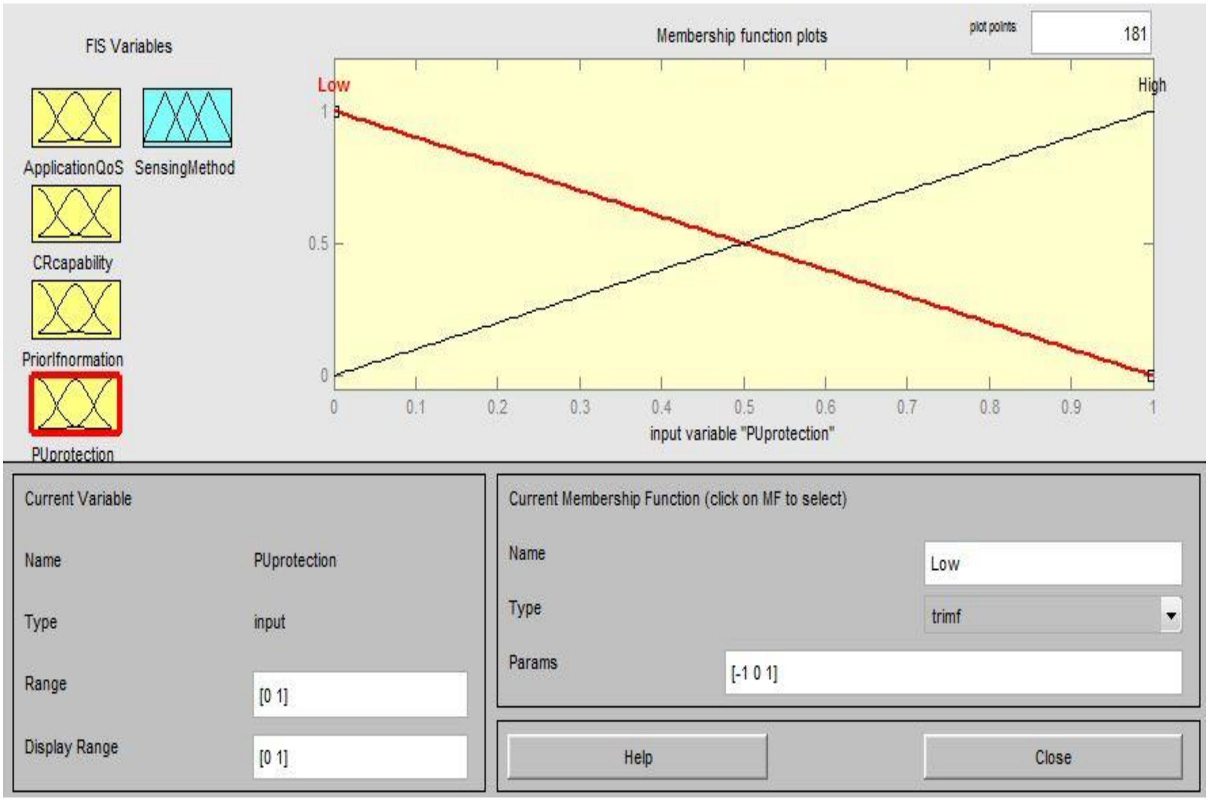
Appendix A 2 Input membership function for CR capability



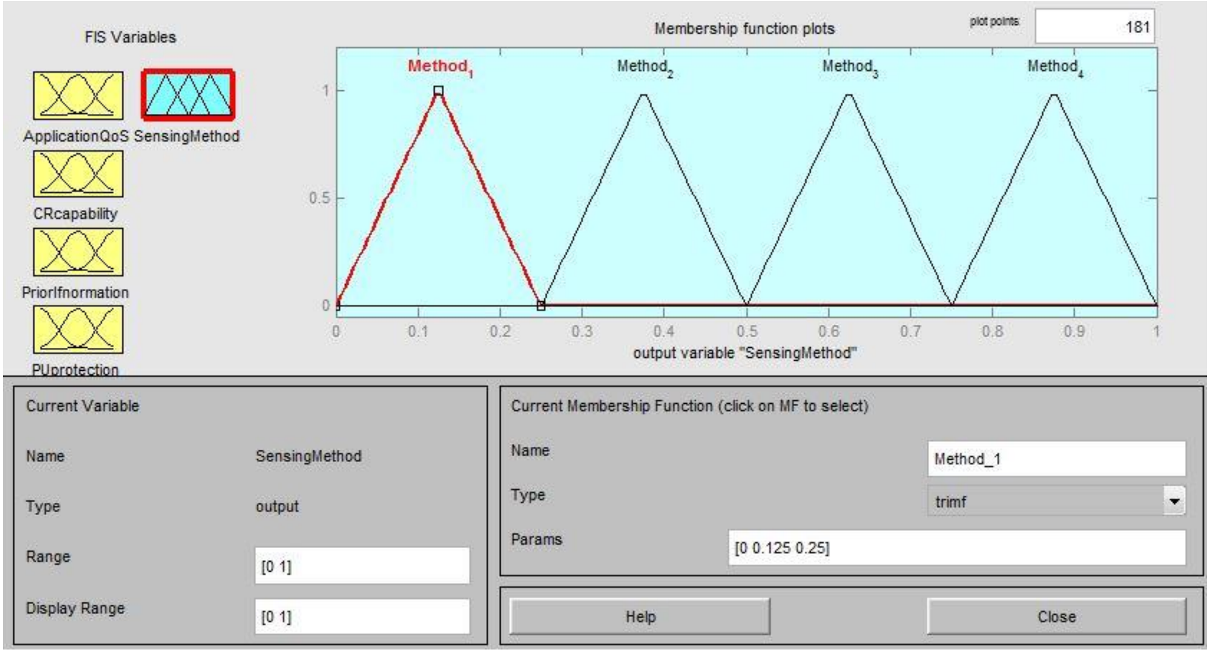
Appendix A 3 Input membership function for Prior information



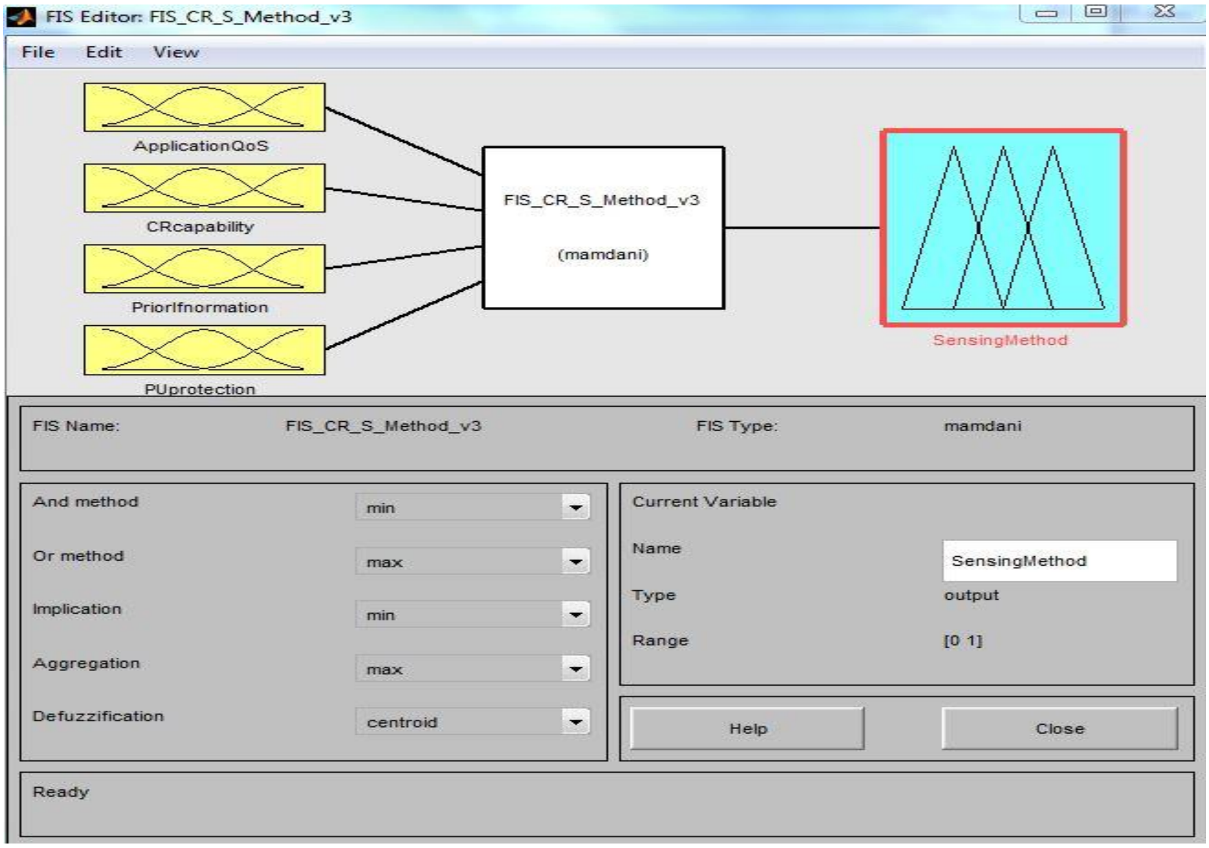
Appendix A 4 Input membership function for PU protection



Appendix A 5 Output membership function Sensing Methods



Appendix A 6 Fuzzy Inference System (FIS) properties settings



Appendix A 7 FIS code in MATLAB

```
[System]

Name='FIS_CR_S_Method_v3'

Type='mamdani'

Version=2.0

NumInputs=4

NumOutputs=1

NumRules=73

AndMethod='min'

OrMethod='max'

ImpMethod='min'

AggMethod='max'

DefuzzMethod='centroid'



[Input1]

Name='ApplicationQoS'

Range=[0 1]

NumMFs=4

MF1='Low':'trimf',[-0.3333 0 0.3333]

MF2='Med':'trimf',[0 0.3333 0.6667]
```

```
MF3='High':'trimf',[0.3333 0.6667 1]

MF4='VeryHigh':'trimf',[0.6667 1 1.333]


[Input2]

Name='CRcapability'

Range=[0 1]

NumMFs=3

MF1='low':'trimf',[-0.5 0 0.5]

MF2='Med':'trimf',[0 0.5 1]

MF3='High':'trimf',[0.5 1 1.5]


[Input3]

Name='PriorIfnormation'

Range=[0 1]

NumMFs=3

MF1='Low':'trimf',[-0.5 0 0.5]

MF2='Med':'trimf',[0 0.5 1]

MF3='High':'trimf',[0.5 1 1.5]


[Input4]
```

```
Name='PUprotection'

Range=[0 1]

NumMFs=2

MF1='Low':'trimf',[-1 0 1]

MF2='High':'trimf',[0 1 2]


[Output1]

Name='SensingMethod'

Range=[-0.1 1.1]

NumMFs=4

MF1='Method_1':'trimf',[-0.1666 0 0.1666]

MF2='Method_2':'trimf',[0.1667 0.3333 0.5]

MF3='Method_3':'trimf',[0.5 0.6666 0.8333]

MF4='Method_4':'trimf',[0.8333 1 1.1666]


[Rules]

4 3 3 2, 2 (1) : 1

4 3 3 1, 4 (1) : 1

4 3 2 2, 2 (1) : 1

4 3 2 1, 2 (1) : 1
```

4 2 2 2, 2 (1) : 1

4 2 2 1, 2 (1) : 1

4 3 1 2, 3 (1) : 1

4 3 1 1, 3 (1) : 1

4 2 1 2, 3 (1) : 1

4 2 1 1, 3 (1) : 1

4 2 3 2, 2 (1) : 1

4 2 3 1, 2 (1) : 1

4 1 3 2, 3 (1) : 1

4 1 2 2, 3 (1) : 1

4 1 1 2, 3 (1) : 1

4 1 3 1, 4 (1) : 1

4 1 2 1, 4 (1) : 1

4 1 1 1, 4 (1) : 1

3 3 3 2, 2 (1) : 1

3 3 3 1, 3 (1) : 1

3 3 2 2, 3 (1) : 1

3 3 2 1, 3 (1) : 1

3 3 1 2, 3 (1) : 1

3 3 1 1, 3 (1) : 1

3 2 2 2, 3 (1) : 1

3 2 2 1, 3 (1) : 1

3 2 1 2, 3 (1) : 1

3 2 1 1, 3 (1) : 1

3 2 3 2, 3 (1) : 1

3 2 3 1, 3 (1) : 1

3 1 3 2, 3 (1) : 1

3 1 2 2, 3 (1) : 1

3 1 1 2, 3 (1) : 1

3 1 3 1, 4 (1) : 1

3 1 2 1, 4 (1) : 1

3 1 1 1, 4 (1) : 1

2 3 3 2, 1 (1) : 1

2 3 3 1, 1 (1) : 1

2 3 2 2, 2 (1) : 1

2 3 2 1, 2 (1) : 1

2 3 1 2, 3 (1) : 1

2 3 1 1, 3 (1) : 1

2 2 3 2, 2 (1) : 1

2 2 3 1, 2 (1) : 1

2 2 2 2, 2 (1) : 1

2 2 2 1, 2 (1) : 1

2 2 1 2, 3 (1) : 1

2 2 1 1, 3 (1) : 1

2 1 3 2, 3 (1) : 1

2 1 2 2, 3 (1) : 1

2 1 1 2, 3 (1) : 1

2 1 3 1, 4 (1) : 1

2 1 2 1, 4 (1) : 1

2 1 1 1, 4 (1) : 1

1 3 3 2, 1 (1) : 1

1 3 3 1, 1 (1) : 1

1 3 2 2, 1 (1) : 1

1 3 2 1, 1 (1) : 1

1 2 3 2, 2 (1) : 1

1 2 3 1, 2 (1) : 1

1 2 2 2, 2 (1) : 1

1 2 2 1, 2 (1) : 1

1 3 1 2, 3 (1) : 1

1 2 1 2, 3 (1) : 1

1 1 1 2, 3 (1) : 1

1 3 1 1, 4 (1) : 1

1 2 1 1, 4 (1) : 1

1 1 1 1, 4 (1) : 1

1 1 3 2, 3 (1) : 1

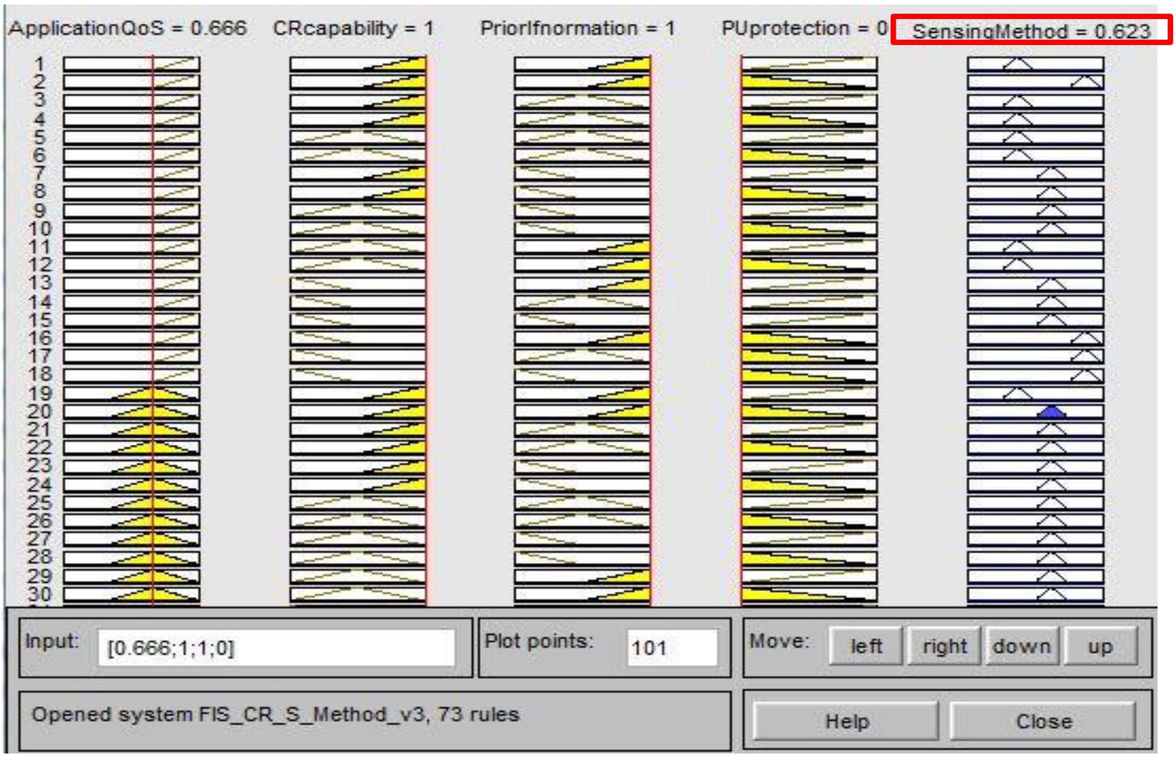
1 1 2 2, 3 (1) : 1

1 1 3 1, 4 (1) : 1

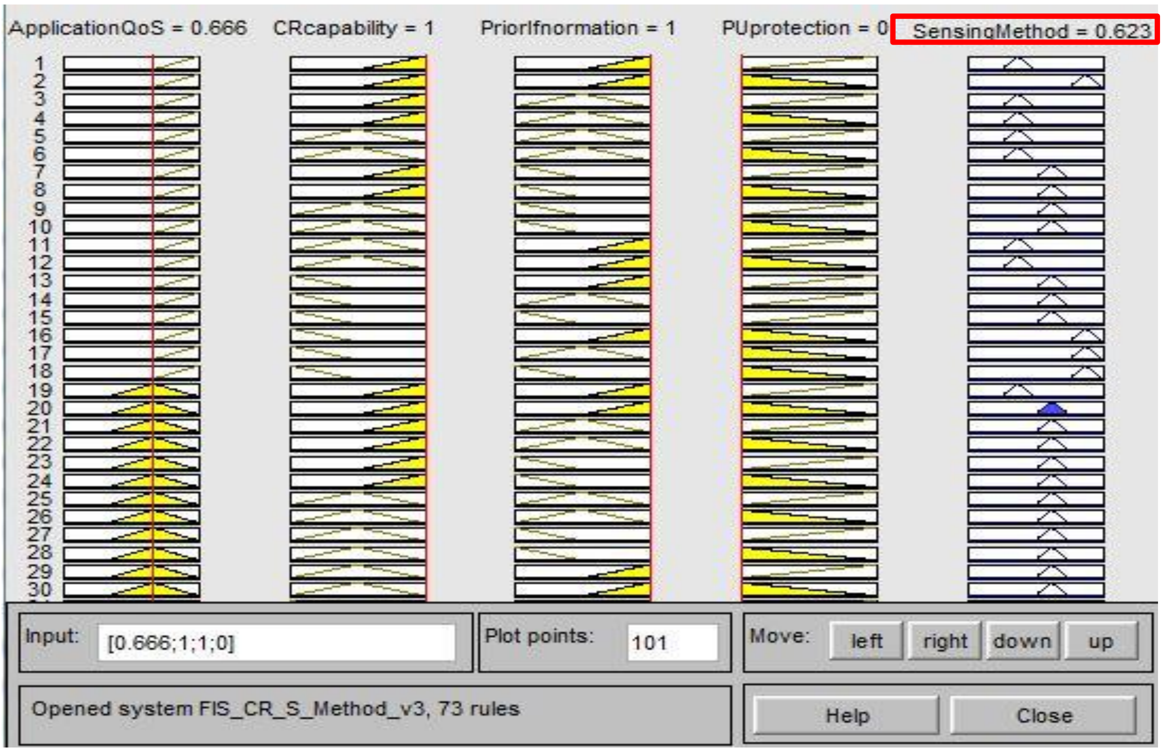
1 1 2 1, 4 (1) : 1

1 1 1 1, 4 (1) : 1

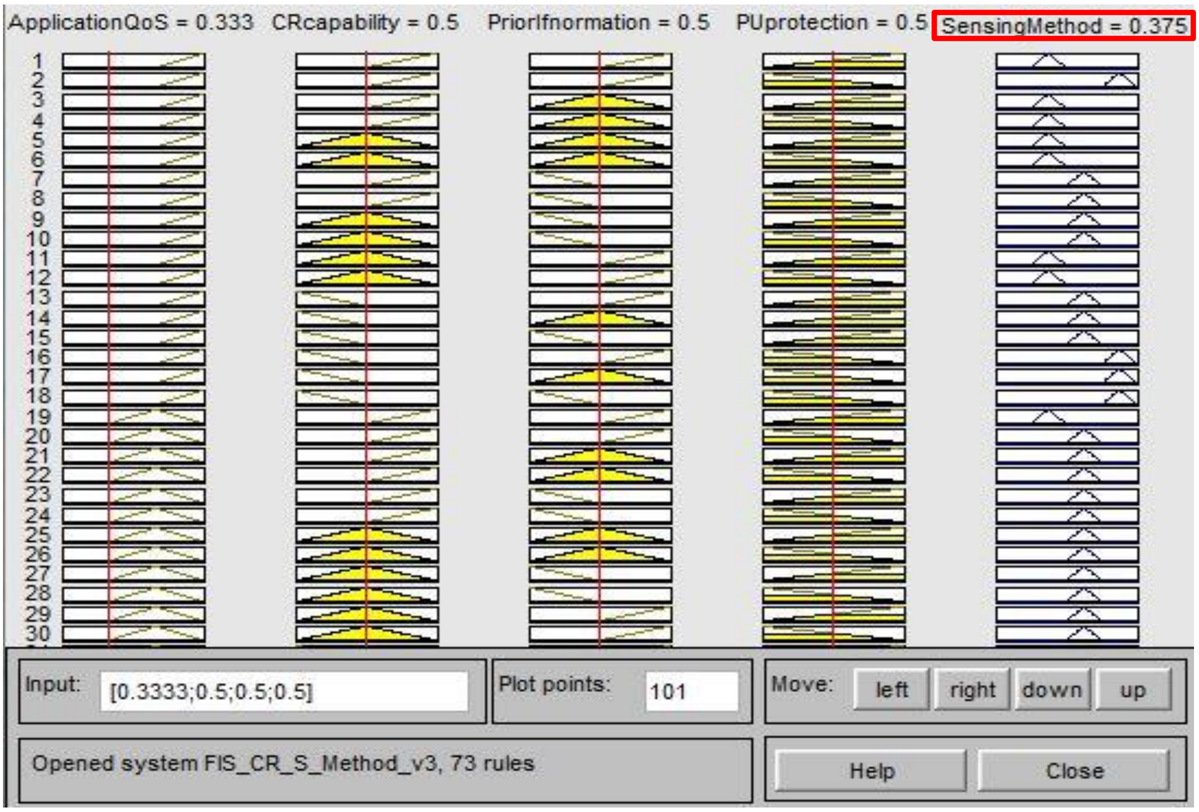
Appendix A 8 FIS output for input variables [1; 1; 1; 0]



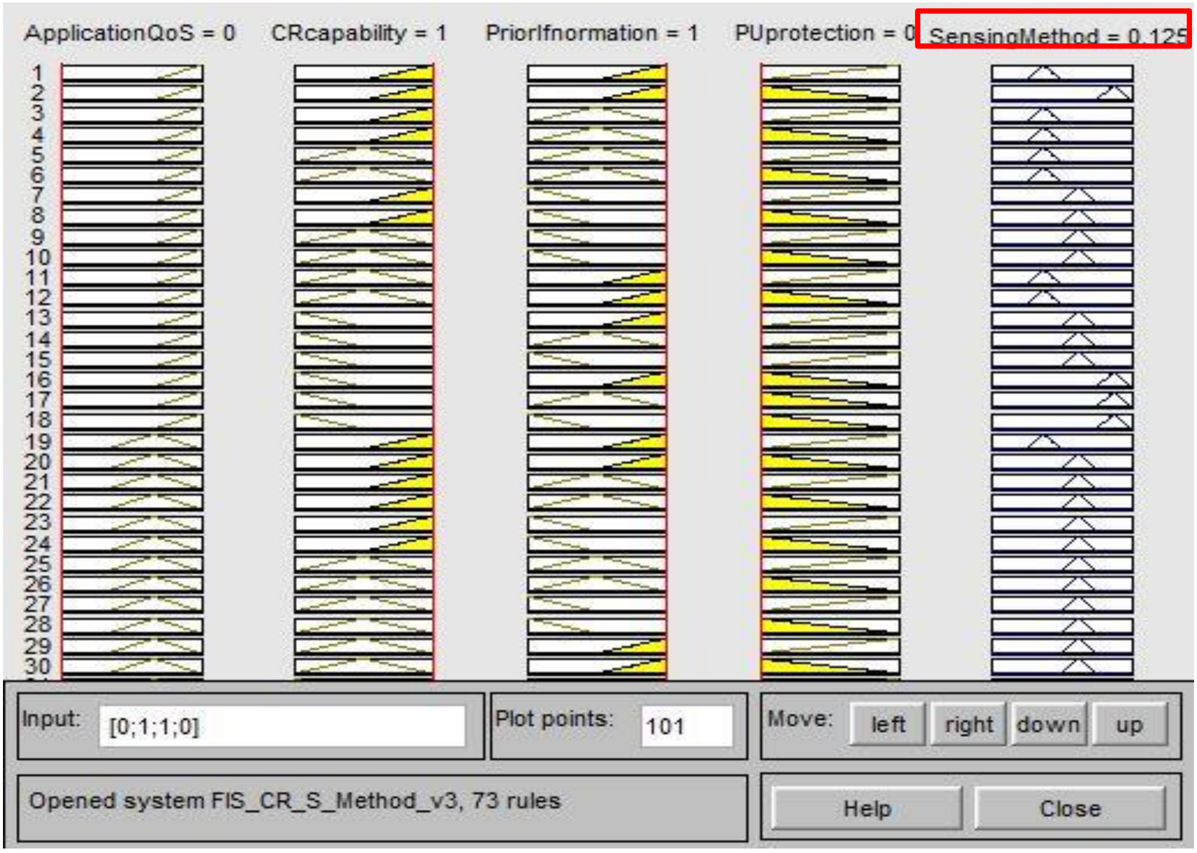
Appendix A 9 FIS output for input variables [0.666; 1; 1; 0]



Appendix A 10 FIS output for input variables [0.333; 1; 1; 0]

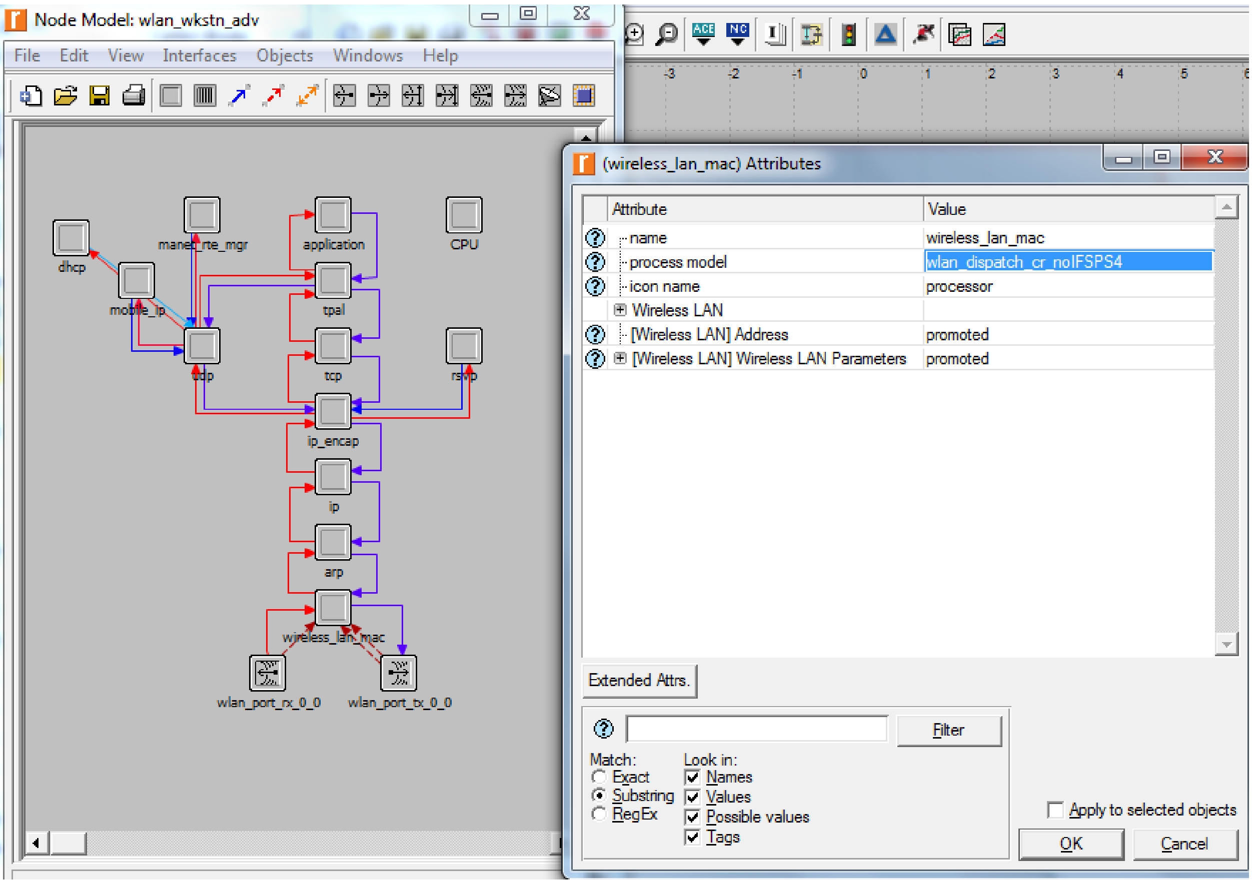


Appendix A 11 FIS output for input variables [0; 1; 1; 0]

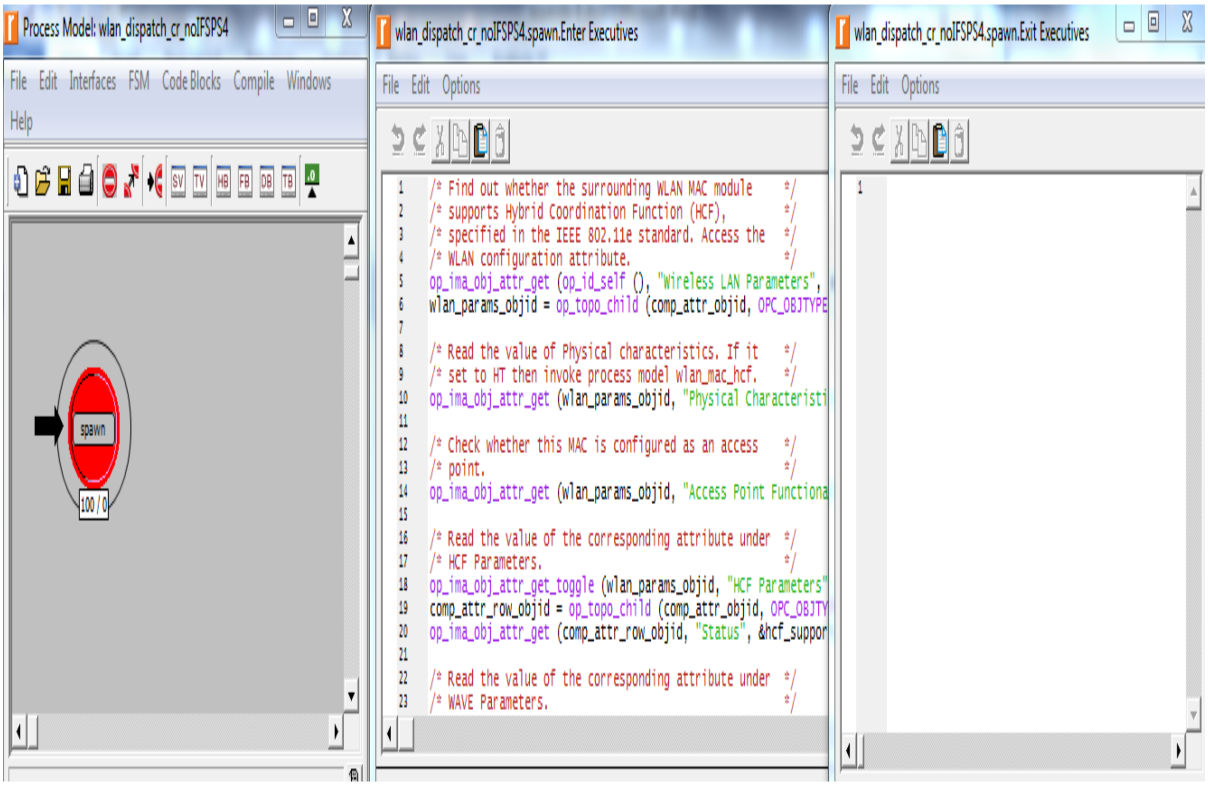


Appendix B: Riverbed Modeler codes and snapshots

Appendix B 1 Advanced WLAN node model (change the MAC process model to one of the customised models based on the required sensing strategy)



Appendix B 2 Snapshots of ‘wlan_dispatch_cr_noIFSPS4’ as an example for MAC process model



Appendix B 3 Code for ‘wlan_dispatch_cr_noIFSPS4’ (Enter Excutive) as an example because all other nodes use the same code except the child process model (highlited) to be changed accordingly

```
/* Find out whether the surrounding WLAN MAC module              */

/* supports Hybrid Coordination Function (HCF),                  */

/* specified in the IEEE 802.11e standard. Access the       */

/* WLAN configuration attribute.                                       */

op_ima_obj_attr_get (op_id_self (), ‘Wireless LAN Parameters’, &comp_attr_objid);

wlan_params_objid = op_topo_child (comp_attr_objid, OPC_OBJTYPE_GENERIC, 0);



/* Read the value of Physical characteristics. If it     */
```

```
/* set to HT then invoke process model wlan_mac_hcf.       */

op_ima_obj_attr_get (wlan_params_objid, 'Physical Characteristics', &phy_char);



/* Check whether this MAC is configured as an access         */

/* point.
          */

op_ima_obj_attr_get (wlan_params_objid, 'Access Point Functionality',
&ap_functionality);



/* Read the value of the corresponding attribute under       */

/* HCF Parameters.
          */

op_ima_obj_attr_get_toggle (wlan_params_objid, 'HCF Parameters', &comp_attr_objid);

comp_attr_row_objid = op_topo_child (comp_attr_objid, OPC_OBJTYPE_GENERIC, 0);

op_ima_obj_attr_get (comp_attr_row_objid, 'Status', &hcf_support_int);



/* Read the value of the corresponding attribute under       */

/* WAVE Parameters.
          */

op_ima_obj_attr_get (wlan_params_objid, 'WAVE Parameters', &comp_attr_objid);

comp_attr_row_objid = op_topo_child (comp_attr_objid, OPC_OBJTYPE_GENERIC, 0);
```

```
op_ima_obj_attr_get_toggle (comp_attr_row_objid, 'WAVE Functionality',
&wave_functionality);


/* Bit shift the phy char.                                         */

phy_char = (1 << (phy_char + WLANC_PHY_CHAR_BIT_SHIFT));


/* Create the appropriate MAC process model. If the HT    */

/* PHY type is set, we will create HCF MAC as support      */

/* of HCF is mandatory as per the 802.11n standard              */

/* section 5.2.9. Similarly, WAVE functionality is            */

/* supported only with HCF MAC.                                    */

if ((hcf_support_int == OPC_BOOLINT_ENABLED)             ||

        (wave_functionality == OPC_BOOLINT_ENABLED && ap_functionality ==
OPC_BOOLINT_DISABLED) ||

        (phy_char & WlanC_HT_OFDM_11n)
                                          )

        {

        /* Create HCF process.
        */

        mac_prohandle = op_pro_create ('wlan_mac_hcf_cr_noIFSPS4', OPC_NIL);
```

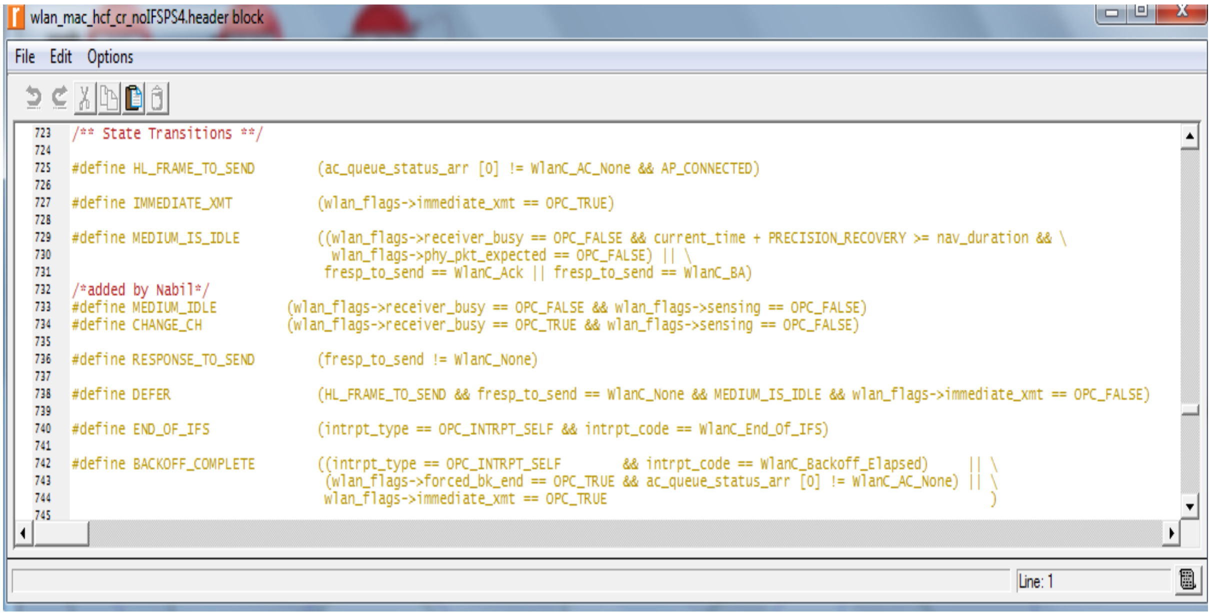
```
/* If the physical characteristics are set to HT           */

/* or WAVE operation is enabled, but the HCF                    */

/* support is disabled write a log message to warn   */

/* the user.
*/

if (hcf_support_int == OPC_BOOLINT_DISABLED)

        {

        /* Register the log handles and related flags.           */

        config_log_handle      =                               op_prg_log_handle_create
(OpC_Log_Category_Configuration, 'Wireless LAN', 'MAC Configuration', 128);



        if (phy_char & WlanC_HT_OFDM_11n)

                {

                /* Write the warning message for HT.                     */

                wlan_phy_char_to_str (phy_char, phy_char_str);

                op_prg_log_entry_write (config_log_handle,

                        'WARNING:\n'

                        ' HCF (802.11e) support is set to \'Not Supported\' in the
WLAN MAC with physical characteristics set to %s\n'

                        '\n'

                        'ACTION:\n'
```

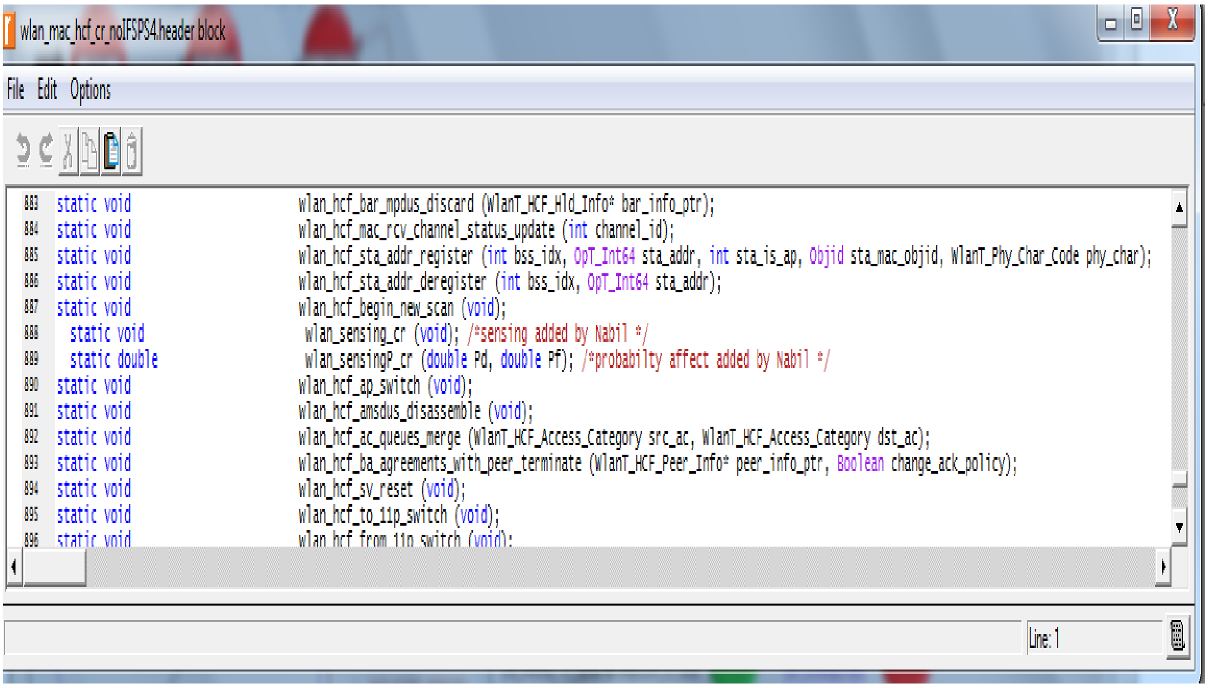
```
                    ' The HCF support is enabled in the WLAN MAC.\n'
                    '\n'
                    'CAUSE:\n'
                    ' All STAs that support IEEE 802.11n physical characteristics must be capable of supporting HCF (802.11e)'
                    ' as required by the IEEE 802.11n-2009 standard.\n'
                    '\n'
                    'SUGGESTION:\n'
                    ' Ensure that all WLAN MACs with IEEE 802.11n physical characteristics have HCF support enabled.\n', phy_char_str);
                }
        else if (wave_functionality == OPC_BOOLINT_ENABLED)
                {
                /* Write the warning message for WAVE.            */
                op_prg_log_entry_write (config_log_handle,
                    'WARNING:\n'
                    ' HCF (802.11e) support is set to \'Not Supported\' in the WLAN MAC even though WAVE (11p) operation'
                    ' is enabled for the same MAC.\n'
                    '\n'
                    'ACTION:\n'
```

```
                                        ' The HCF support is enabled in the WLAN MAC.\n'
                                        '\n'
                                        'CAUSE:\n'
                                        ' WAVE operation is supported only in QSTAs (11e-capable
WLAN entities).\n'
                                        '\n'
                                        'SUGGESTION:\n'
                                        ' Ensure that all WLAN MACs that are WAVE capable are also
HCF capable.\n');
                              }
                    }
          }
else
          mac_prohandle = op_pro_create ('wlan_mac_cr', OPC_NIL);


/* Make the child process the recipient of the                              */
/* interrupts of the module.                                                */
op_intrpt_type_register (OPC_INTRPT_STRM,  mac_prohandle);
op_intrpt_type_register (OPC_INTRPT_STAT,  mac_prohandle);
op_intrpt_type_register (OPC_INTRPT_REMOTE, mac_prohandle);
```

```
/* Spawn the MAC child process.                                    */

op_pro_invoke (mac_prohandle, OPC_NIL);
```

Appendix B 4 Codes and settings shared by all customised 'wlan_mac_hcf ' child process model, snapshots of 'wlan_mac_hcf_cr_noIFSPS4' as example





wlan_mac_hcf_cr_noIFSPS4.state variables

	Type	Name	Comments
204	Sbhandle	ampdu_reassembly_buffer	/* Reassembly buffer to extract MPDUs from A-MPDU frame. */
205	Stathandle	mpdus_per_ppdu_stathandle[WLANC_HCF_TC_COUNT]	/* Stathandle for number of MPDUs per PPDU. */
206	double	frame_transmission_end_time	/* Time when the transmission of a frame ends. */
207	double	pifs_recovery_wait_time	/* The time the STA is required to wait to carry out a PIFS recovery. */
208	int	num_stalled_packets[WLANC_HCF_AC_COUNT]	/* Number of packets of a BA agreement that are stalled for transmission. */
209	int	bar_fail_ssn	/* Stores the starting sequence number of the failed BAR. */
210	WlanT_11p_Operation	dot11p_mode	/* 802.11p (WAVE) capability and operation mode information. */
211	WlanT_WAVE_Config*	wave_config_ptr	/* Storage for 11p specific attribute values. This will be populated */
212	WlanT_WAVE_Config*	phy_tech_info_ptr	/* Storage for attribute values that are specific to configured PHY */
213	double	sensing_duration	/* sensing time duration added by NABIL */
214	Evhandle	sensing_timeout_evh	/* sensing timeout added by Nabil */
215	OsysT_Random_Stream_Handle*	gen_ptr	/* Handle random stream to generate random number by Nabil */
216	PrgT_Random_Gen*	my_rng	/* Handle random stream to generate random number from OP functions by Nabil */
217	double	rand_dbl	/* Random double number from 0 to 1.0 added by Nabil */
218			

Find: Next Previous Match case Match words

Edit ASCII Delete OK Cancel

wlan_mac_hcf_cr_noIFSPS4.temporary variables

File Edit Options

```
OmsT_Pr_Handle          process_record_handle;
List*                   proc_record_handle_list_ptr;
OpT_Int64*              sta_addr_ptr;
int                     record_handle_list_size;
int                     ap_count;
OpT_Int64               sta_addr;
int                     statype;
Boolean                 ap_ba_support;
Boolean                 ht_ap_legacy_support;
Objid                   mac_objid;
Objid                   mac_if_module_objid;
Objid                   parent_subnet_objid;
char                    name_str [128];
Objid                   params_attr_objid;
Objid                   wlan_params_comp_attr_objid;
int                     addr_index;
int                     num_forty_capable_ht_stas;
Prohandle               own_prohandle;
WlanT_HCF_Peer_Info*    peer_info_ptr;
void*                   dummy_ptr;
char                    msg1 [256];
WlanT_Phy_Char_Code     sta_phy_char_flag, ap_phy_char_flag;
WlanT_Ht_Info*          sta_ht_info_ptr;
WlanT_Ht_Info*          ap_ht_info_ptr;
Boolean                 bad_packet_rcvd = OPC_FALSE;
Boolean                 bad_cts_to_self_rcvd = OPC_FALSE;
Boolean                 backoff_on_secondary_busy = OPC_FALSE;
Boolean                 pifs_failure_on_sec_busy = OPC_FALSE;
Boolean                 previous_secondary_ch_busy_flag = OPC_FALSE;
double                  tx_power;
double                  x_pos, y_pos, z_pos;
double                  inter_frame_spacing;
double                  time_left_txop;
char                    phy_char_str[128];

Packet*                 pkptr;

int                         ac_total_slots, min_backoff_slots, max_cw_slots;
WlanT_HCF_Access_Category   ac, min_backoff_ac, max_cw_ac;
int                         slots_completed, ac_slots_completed;
WlanT_HCF_Access_Category   new_hlpk_ac;

Boolean                     state_reentered = OPC_FALSE;
int                         i, j;
int                         integer_mac_address = -1;
```

Line: 1

wlan_mac_hcf_cr_noIFSPS4.function block

File Edit Options

```
    /* Initially resetting all the flags.                                   */
    wlan_flags->rts_sent                = OPC_FALSE;
    wlan_flags->rcvd_bad_packet         = OPC_FALSE;
    wlan_flags->bad_packet_dropped      = OPC_FALSE;
    wlan_flags->receiver_busy           = OPC_FALSE;
    wlan_flags->phy_pkt_expected        = OPC_FALSE;
    wlan_flags->transmitter_busy        = OPC_FALSE;
    wlan_flags->gateway_flag            = OPC_FALSE;
    wlan_flags->bridge_flag             = OPC_FALSE;
    wlan_flags->wait_eifs_dur           = OPC_FALSE;
    wlan_flags->immediate_xmt           = OPC_FALSE;
    wlan_flags->forced_bk_end           = OPC_FALSE;
    wlan_flags->nav_updated             = OPC_FALSE;

    wlan_flags->tx_beacon               = OPC_FALSE;

    wlan_flags->non_erp_present         = OPC_FALSE;
    wlan_flags->wait_signal_ext         = OPC_FALSE;
    wlan_flags->rcvd_bad_cts            = OPC_FALSE;
    wlan_flags->bad_cts_dropped         = OPC_FALSE;
    wlan_flags->edca_params_updated     = OPC_FALSE;
    wlan_flags->txop_on                 = OPC_FALSE;
    wlan_flags->not_first_in_txop       = OPC_FALSE;
    wlan_flags->scanning                = OPC_FALSE;
    /*added by Nabil*/
    wlan_flags->sensing             = OPC_FALSE;

    wlan_flags->txop_limit_reserved     = OPC_FALSE;
    wlan_flags->txop_start              = OPC_FALSE;
    wlan_flags->pifs_recovery           = OPC_FALSE;
    wlan_flags->jam_rxst_before_txend   = OPC_FALSE;

    /* IEEE 802.11n flags.                                                  */
    wlan_flags->non_gf_ht_sta_present   = OPC_FALSE;
    wlan_flags->non_ht_sta_present      = OPC_FALSE;
    wlan_flags->rifs_separated_tx       = OPC_FALSE;
    wlan_flags->forty_mhz_operation     = OPC_FALSE;
    wlan_flags->forty_mhz_txop          = OPC_FALSE;
    wlan_flags->non_fc_ht_sta_present   = OPC_FALSE;
    wlan_flags->secondary_ch_busy       = OPC_FALSE;
    wlan_flags->is_tx_40mhz_ppdu        = OPC_FALSE;

    /* Similarly allocate memory for the AC specific bitmap flags and       */
    /* initialize them by resetting all the bits.                           */
    wlan_ac_flags = (WlanT_AC_Bit_Flags *) op_prg_mem_alloc (sizeof (WlanT_AC_Bit_Flags));
    wlan_ac_flags->cw_required      = 0;
    wlan_ac_flags->frsize_req_rts   = 0;
    wlan_ac_flags->std_cwmin        = 0;
    wlan_ac_flags->std_cwmax        = 0;
    wlan_ac_flags->stats_registered = 0;
    wlan_ac_flags->implicit_bar_sent = 0;

    /* Set the flag corresponding to optional 802.11g/11n protection        */
```

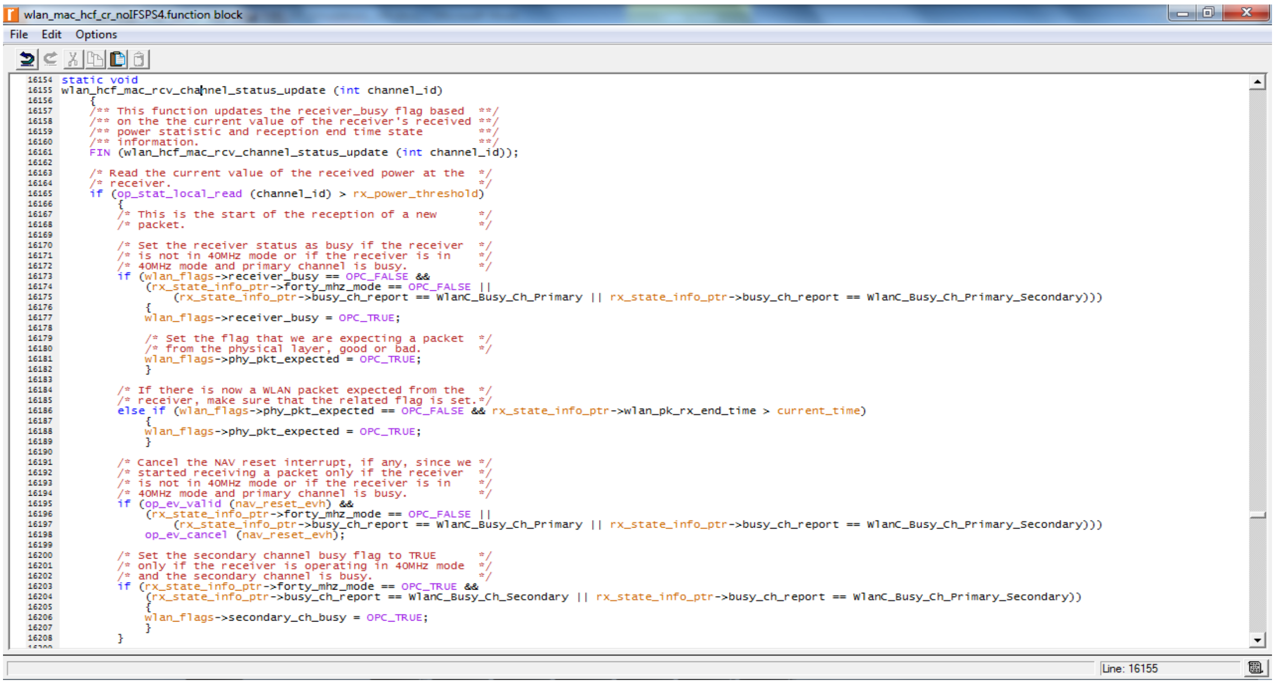
Line: 323

wlan_mac_hcf_cr_noIFSPS4.function block

File Edit Options

```
        }

    /* Else a packet reception is complete. Check whether   */
    /* the receiver became available while it was busy. It  */
    /* may not have been busy if we were receiving a noise  */
    /* packet with a weak signal.                           */
    else if (wlan_flags->receiver_busy || wlan_flags->secondary_ch_busy)
        {
        /* Check if the primary channel was busy.           */
        if (wlan_flags->receiver_busy)
            {
            /* Compare the receiver's reception end time    */
            /* value with the current time to determine     */
            /* its status.                                  */
            if (rx_state_info_ptr->rx_end_time - PRECISION_RECOVERY <= current_time)
                {
                wlan_flags->receiver_busy = OPC_FALSE;
                }

            /* Update the receiver idle time with current   */
            /* time. Even though the receiver can be still  */
            /* busy (in case of a collision), this is       */
            /* necessary to note the completion time of the */
            /* last reception.                              */
            rcv_idle_time = current_time;
            }

        /* Check if the packet was interfering with the     */
        /* secondary channel as well. If so then update the */
        /* secondary channel idle time.                     */
        if (wlan_flags->secondary_ch_busy)
            {
            /* Compare the receiver's reception end time    */
            /* value with the current time to determine     */
            /* its status.                                  */
            if (rx_state_info_ptr->secch_rx_end_time - PRECISION_RECOVERY <= current_time)
                {
                wlan_flags->secondary_ch_busy = OPC_FALSE;
                }

            /* Update the receiver idle time with current   */
            /* time. Even though the receiver can be still  */
            /* busy (in case of a collision), this is       */
            /* necessary to note the completion time of the */
            /* last reception.                              */
            secondary_ch_idle_time = current_time;
            }
        }

    FOUT;
    }

static void
wlan_hcf_sta_addr_register (int bssid, OpT_Int64 sta_addr, int sta_is_ap, Objid sta_mac_objid, WlanT_Phy_Char_Code
```

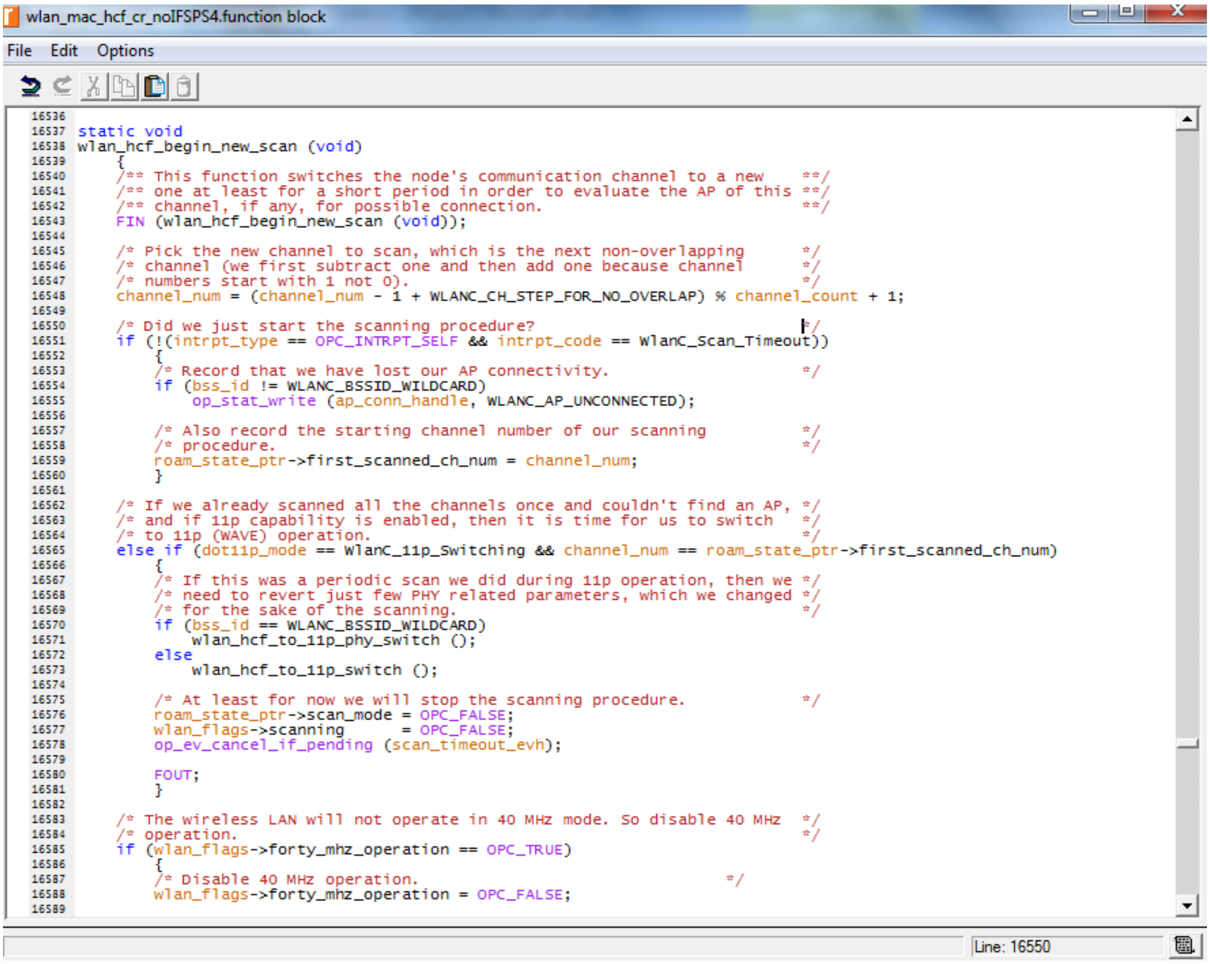
Line: 16155


wlan_mac_hcf_cr_noIFSPS4.function block

File Edit Options

```
static void
wlan_hcf_mac_rcv_channel_status_update (int channel_id)
    {
    /** This function updates the receiver_busy flag based  **/
    /** on the the current value of the receiver's received **/
    /** power statistic and reception end time state        **/
    /** information.                                        **/
    FIN (wlan_hcf_mac_rcv_channel_status_update (int channel_id));

    /* Read the current value of the received power at the  */
    /* receiver.                                            */
    if (op_stat_local_read (channel_id) > rx_power_threshold)
        {
        /* This is the start of the reception of a new      */
        /* packet.                                          */

        /* Set the receiver status as busy if the receiver  */
        /* is not in 40MHz mode or if the receiver is in    */
        /* 40MHz mode and primary channel is busy.          */
        if (wlan_flags->receiver_busy == OPC_FALSE &&
            (rx_state_info_ptr->forty_mhz_mode == OPC_FALSE ||
                (rx_state_info_ptr->busy_ch_report == WlanC_Busy_Ch_Primary || rx_state_info_ptr->busy_ch_report == WlanC_Busy_Ch_Primary_Secondary)))
            {
            wlan_flags->receiver_busy = OPC_TRUE;

            /* Set the flag that we are expecting a packet  */
            /* from the physical layer, good or bad.        */
            wlan_flags->phy_pkt_expected = OPC_TRUE;
            }

        /* If there is now a WLAN packet expected from the  */
        /* receiver, make sure that the related flag is set.*/
        else if (wlan_flags->phy_pkt_expected == OPC_FALSE && rx_state_info_ptr->wlan_pk_rx_end_time > current_time)
            {
            wlan_flags->phy_pkt_expected = OPC_TRUE;
            }

        /* Cancel the NAV reset interrupt, if any, since we */
        /* started receiving a packet only if the receiver  */
        /* is not in 40MHz mode or if the receiver is in    */
        /* 40MHz mode and primary channel is busy.          */
        if (op_ev_valid (nav_reset_evh) &&
            (rx_state_info_ptr->forty_mhz_mode == OPC_FALSE ||
                (rx_state_info_ptr->busy_ch_report == WlanC_Busy_Ch_Primary || rx_state_info_ptr->busy_ch_report == WlanC_Busy_Ch_Primary_Secondary)))
            op_ev_cancel (nav_reset_evh);

        /* Set the secondary channel busy flag to TRUE      */
        /* only if the receiver is operating in 40MHz mode  */
        /* and the secondary channel is busy.               */
        if (rx_state_info_ptr->forty_mhz_mode == OPC_TRUE &&
            (rx_state_info_ptr->busy_ch_report == WlanC_Busy_Ch_Secondary || rx_state_info_ptr->busy_ch_report == WlanC_Busy_Ch_Primary_Secondary))
            {
            wlan_flags->secondary_ch_busy = OPC_TRUE;
            }
        }
```

Line: 16155


wlan_mac_hcf_cr_noIFSPS4.function block

File Edit Options

```
static void
wlan_hcf_begin_new_scan (void)
    {
    /** This function switches the node's communication channel to a new  **/
    /** one at least for a short period in order to evaluate the AP of this **/
    /** channel, if any, for possible connection.                           **/
    FIN (wlan_hcf_begin_new_scan (void));

    /* Pick the new channel to scan, which is the next non-overlapping      */
    /* channel (we first subtract one and then add one because channel      */
    /* numbers start with 1 not 0).                                         */
    channel_num = (channel_num - 1 + WLANC_CH_STEP_FOR_NO_OVERLAP) % channel_count + 1;

    /* Did we just start the scanning procedure?                            */
    if (!(intrpt_type == OPC_INTRPT_SELF && intrpt_code == WlanC_Scan_Timeout))
        {
        /* Record that we have lost our AP connectivity.                    */
        if (bss_id != WLANC_BSSID_WILDCARD)
            op_stat_write (ap_conn_handle, WLANC_AP_UNCONNECTED);

        /* Also record the starting channel number of our scanning          */
        /* procedure.                                                       */
        roam_state_ptr->first_scanned_ch_num = channel_num;
        }

    /* If we already scanned all the channels once and couldn't find an AP, */
    /* and if 11p capability is enabled, then it is time for us to switch   */
    /* to 11p (WAVE) operation.                                             */
    else if (dot11p_mode == WlanC_11p_Switching && channel_num == roam_state_ptr->first_scanned_ch_num)
        {
        /* If this was a periodic scan we did during 11p operation, then we */
        /* need to revert just few PHY related parameters, which we changed */
        /* for the sake of the scanning.                                    */
        if (bss_id == WLANC_BSSID_WILDCARD)
            wlan_hcf_to_11p_phy_switch ();
        else
            wlan_hcf_to_11p_switch ();

        /* At least for now we will stop the scanning procedure.            */
        roam_state_ptr->scan_mode = OPC_FALSE;
        wlan_flags->scanning      = OPC_FALSE;
        op_ev_cancel_if_pending (scan_timeout_evh);

        FOUT;
        }

    /* The wireless LAN will not operate in 40 MHz mode. So disable 40 MHz */
    /* operation.                                                           */
    if (wlan_flags->forty_mhz_operation == OPC_TRUE)
        {
        /* Disable 40 MHz operation.                                */
        wlan_flags->forty_mhz_operation = OPC_FALSE;
```

Line: 16550

wlan_mac_hcf_cr_noIFSPS4.function block

File Edit Options

```
    }

static Boolean
wlan_hcf_ht_seq_trace_header_debug (WlanT_HCF_Hld_Info* hld_ptr)
    {
    char peer_tid_trace[256];
    char msg[256];

    /** This function returns OPC_TRUE if peer specific ODB tracing is      **/
    /** enabled for the destination of the given higher layer packet. If    **/
    /** that is the case, the function also prints header information about **/
    /** the given packet.                                                    **/
    FIN (wlan_hcf_ht_seq_trace_header_debug (hld_ptr));

    if (debug_mode)
        {
        if (wlan_flags->ad_hoc_or_ap)
            sprintf(peer_tid_trace, "wlan_ba_seq_peer_"OPC_INT64_FMT"_tid_%d", hld_ptr->dest_addr, hld_ptr->up);
        else
            sprintf(peer_tid_trace, "wlan_ba_seq_tid_%d", hld_ptr->up);

        if (op_prg_odb_ltrace_active (peer_tid_trace))
            {
            if (wlan_flags->ad_hoc_or_ap)
                sprintf(msg, "TC: [%s], PEER: ["OPC_INT64_FMT"], ACK Policy:[%s]",  WLANC_TC_NAME_ARRAY [hld_ptr->up], hld_ptr->dest_addr,
((tc_config_arr[hld_ptr->up].ba_policy == WlanC_Immediate_BA)? "Immediate":"Delayed"));
            else
                sprintf(msg, "TC: [%s], ACK Policy:[%s]",  WLANC_TC_NAME_ARRAY [hld_ptr->up], (tc_config_arr[hld_ptr->up].ba_policy == WlanC_Immediate_BA)?
"Immediate":"Delayed");
            op_prg_odb_print_major (msg, OPC_NIL);

            FRET (OPC_TRUE);
            }
        }

    FRET (OPC_FALSE);
    }

/*This is the sensing function for adding the effect of that function on the transmission. Added by NABIL */
static void
    wlan_sensing_cr (void)
        {


        /* Initialize the sensing timeout event handle. */

        FIN (wlan_sensing_cr (void));

         wlan_flags->sensing = OPC_TRUE;
         wlan_hcf_mac_rcv_channel_status_update (channel_num);


        FOUT;
          }
```

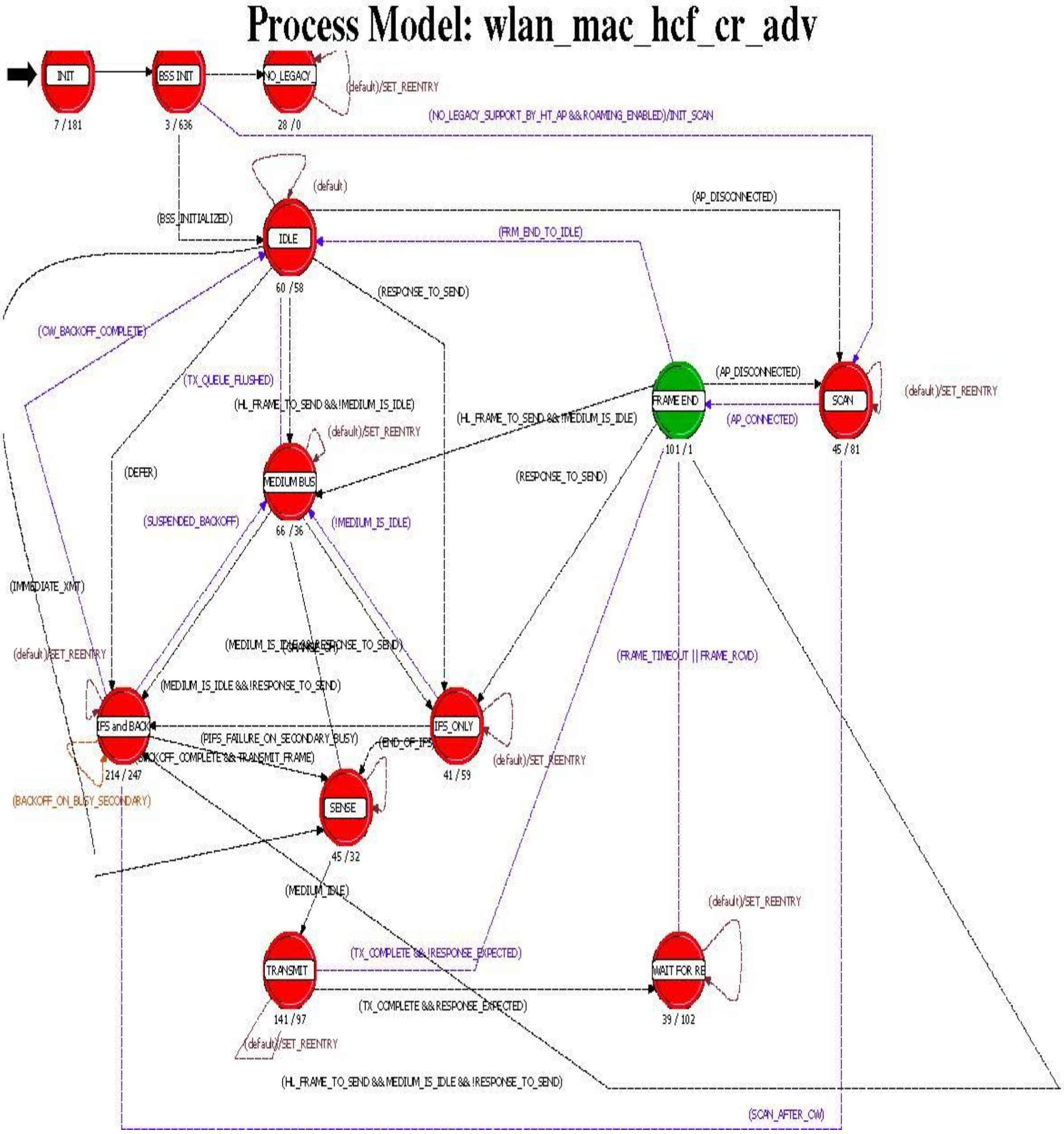
Line: 20570

wlan_mac_hcf_cr_noIFSPS4.function block

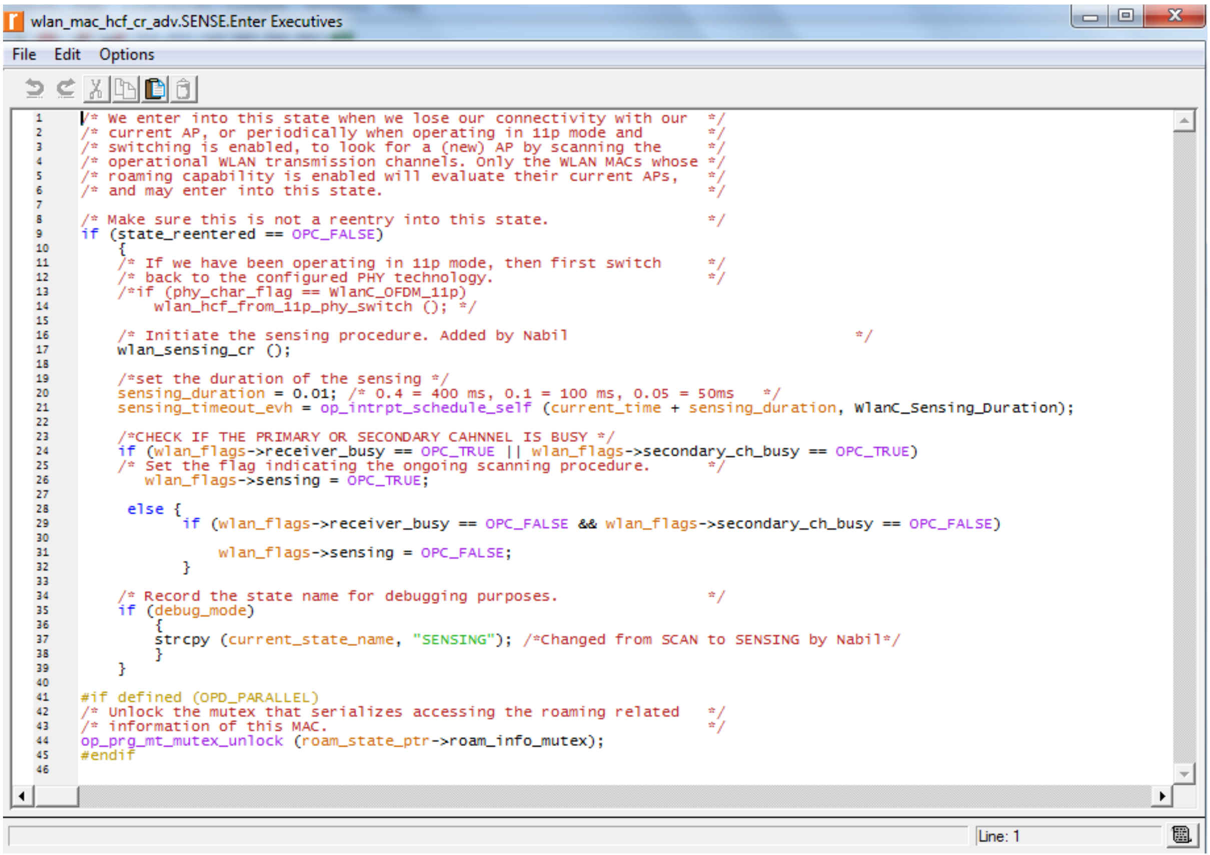
File Edit Options

```
/*this function added to count the probability detection and false alarm in sensing by Nabil */
/*Pd probability of detection, Pf probability of false alarm in sensing function added by Nabil */
static double
wlan_sensingP_cr (double Pd, double Pf)
         {


    FIN (wlan_sensingP_cr (double Pd, double Pf));

         /* create a new random number generator */

   rand_dbl = op_prg_random_double_gen (my_rng);
  // random = rand_dbl;
 op_trace_double ("random", rand_dbl); // added to follow the random value


          /* miss detection */
         if(rand_dbl > Pd) /*(rand_dbl > Pd) */
           {
            if (wlan_flags->receiver_busy == OPC_TRUE || wlan_flags->secondary_ch_busy == OPC_TRUE) // channels are busy
              {
              /*set the busy channel to idel as result of miss detection*/
              wlan_flags->receiver_busy = OPC_FALSE;
              wlan_flags->secondary_ch_busy = OPC_FALSE;
              }
            }


        /* detection */

         if(rand_dbl <= Pd) /*(rand_dbl <= Pd) */
           {
           if (wlan_flags->receiver_busy == OPC_TRUE || wlan_flags->secondary_ch_busy == OPC_TRUE)
           {sensing_duration = sensing_duration + 0.01;}  /* 0.01 = 10ms time required for handover */
           else if ((rand_dbl < Pf) && (wlan_flags->receiver_busy == OPC_FALSE)) //flase alarm
             {
             wlan_flags->receiver_busy = OPC_TRUE;    /* set it to busy becuse of the false alarm */
             sensing_duration = sensing_duration + 0.01;  /* 0.01 = 10ms time required for handover */
             }
           }

         FRET (sensing_duration);
         }
```

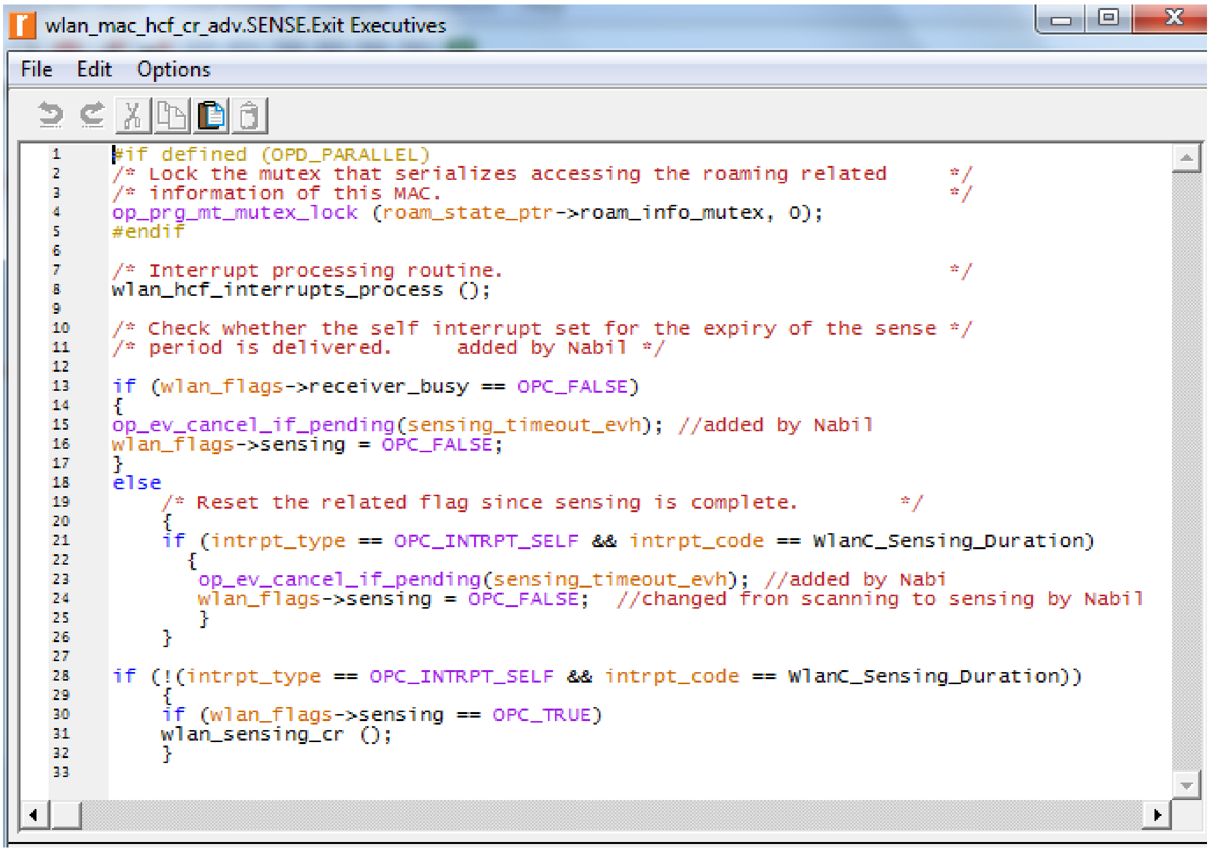
Appendix B 5 Layout for 'wlan_mac_hcf_cr_adv' process model



Appendix B 6 Sense state for ‘wlan_mac_hcf_cr_adv’

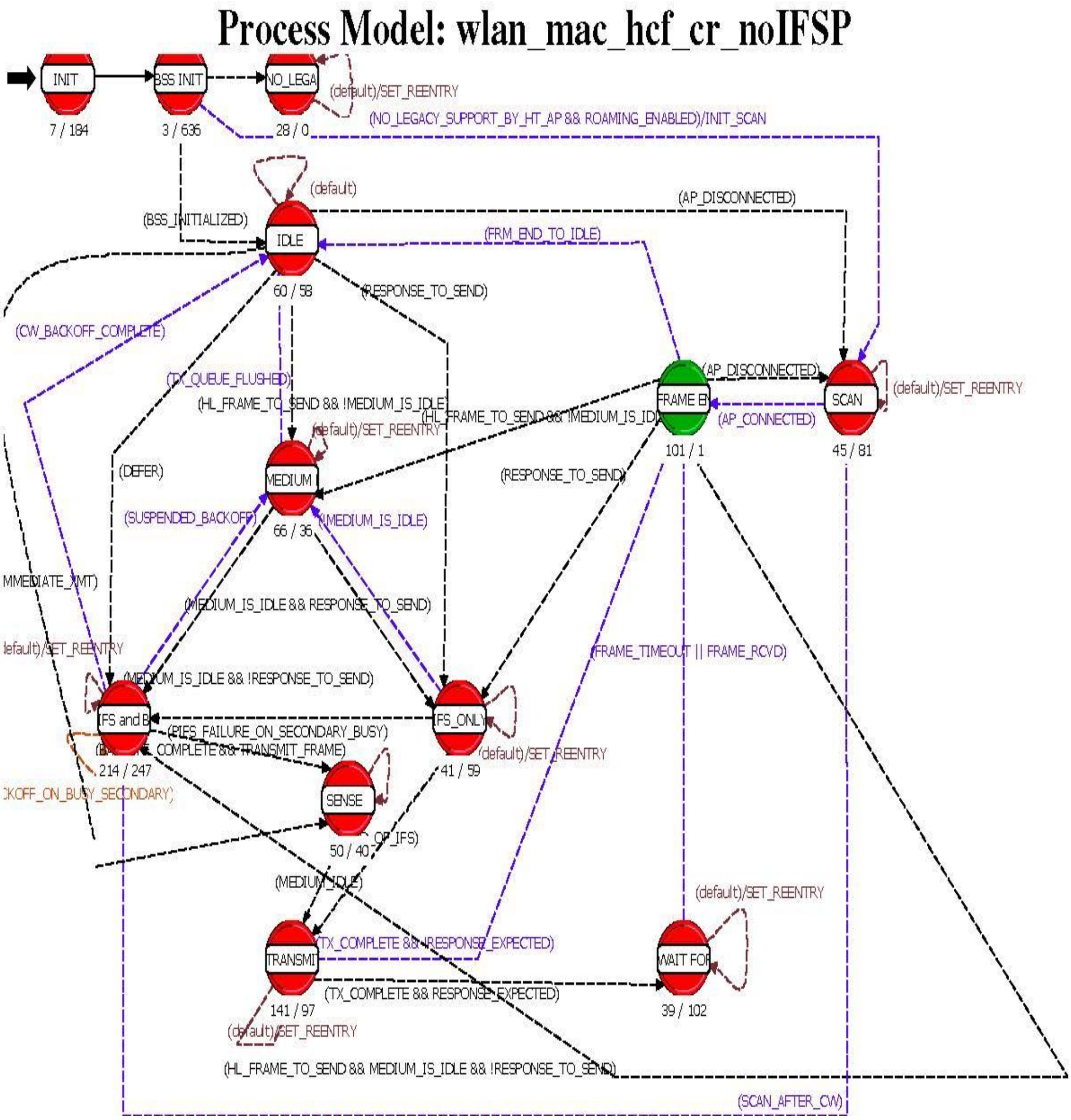



```c
/* We enter into this state when we lose our connectivity with our  */
/* current AP, or periodically when operating in 11p mode and       */
/* switching is enabled, to look for a (new) AP by scanning the     */
/* operational WLAN transmission channels. Only the WLAN MACs whose */
/* roaming capability is enabled will evaluate their current APs,   */
/* and may enter into this state.                                   */

/* Make sure this is not a reentry into this state.                 */
if (state_reentered == OPC_FALSE)
    {
    /* If we have been operating in 11p mode, then first switch     */
    /* back to the configured PHY technology.                       */
    /*if (phy_char_flag == WlanC_OFDM_11p)
        wlan_hcf_from_11p_phy_switch (); */

    /* Initiate the sensing procedure. Added by Nabil                               */
    wlan_sensing_cr ();

    /*set the duration of the sensing */
    sensing_duration = 0.01; /* 0.4 = 400 ms, 0.1 = 100 ms, 0.05 = 50ms   */
    sensing_timeout_evh = op_intrpt_schedule_self (current_time + sensing_duration, WlanC_Sensing_Duration);

    /*CHECK IF THE PRIMARY OR SECONDARY CAHNNEL IS BUSY */
    if (wlan_flags->receiver_busy == OPC_TRUE || wlan_flags->secondary_ch_busy == OPC_TRUE)
    /* Set the flag indicating the ongoing scanning procedure.      */
       wlan_flags->sensing = OPC_TRUE;

     else {
          if (wlan_flags->receiver_busy == OPC_FALSE && wlan_flags->secondary_ch_busy == OPC_FALSE)

              wlan_flags->sensing = OPC_FALSE;
          }

    /* Record the state name for debugging purposes.                */
    if (debug_mode)
        {
        strcpy (current_state_name, "SENSING"); /*Changed from SCAN to SENSING by Nabil*/
        }
    }

#if defined (OPD_PARALLEL)
/* Unlock the mutex that serializes accessing the roaming related   */
/* information of this MAC.                                         */
op_prg_mt_mutex_unlock (roam_state_ptr->roam_info_mutex);
#endif
```




```c
#if defined (OPD_PARALLEL)
/* Lock the mutex that serializes accessing the roaming related     */
/* information of this MAC.                                         */
op_prg_mt_mutex_lock (roam_state_ptr->roam_info_mutex, 0);
#endif

/* Interrupt processing routine.                                    */
wlan_hcf_interrupts_process ();

/* Check whether the self interrupt set for the expiry of the sense */
/* period is delivered.      added by Nabil */

if (wlan_flags->receiver_busy == OPC_FALSE)
{
op_ev_cancel_if_pending(sensing_timeout_evh); //added by Nabil
wlan_flags->sensing = OPC_FALSE;
}
else
    /* Reset the related flag since sensing is complete.        */
    {
    if (intrpt_type == OPC_INTRPT_SELF && intrpt_code == WlanC_Sensing_Duration)
      {
       op_ev_cancel_if_pending(sensing_timeout_evh); //added by Nabi
       wlan_flags->sensing = OPC_FALSE;  //changed fron scanning to sensing by Nabil
      }
    }

if (!(intrpt_type == OPC_INTRPT_SELF && intrpt_code == WlanC_Sensing_Duration))
    {
    if (wlan_flags->sensing == OPC_TRUE)
    wlan_sensing_cr ();
    }
```

Appendix B 7 Layout for 'wlan_mac_hcf_cr_noIFSP' process model



Appendix B 8 Sense state for ‘wlan_mac_hcf_cr_noIFSP’ and ‘wlan_mac_hcf_cr_noIFSPCH’

wlan_mac_hcf_cr_noIFSP.SENSE.Enter Executives

File Edit Options

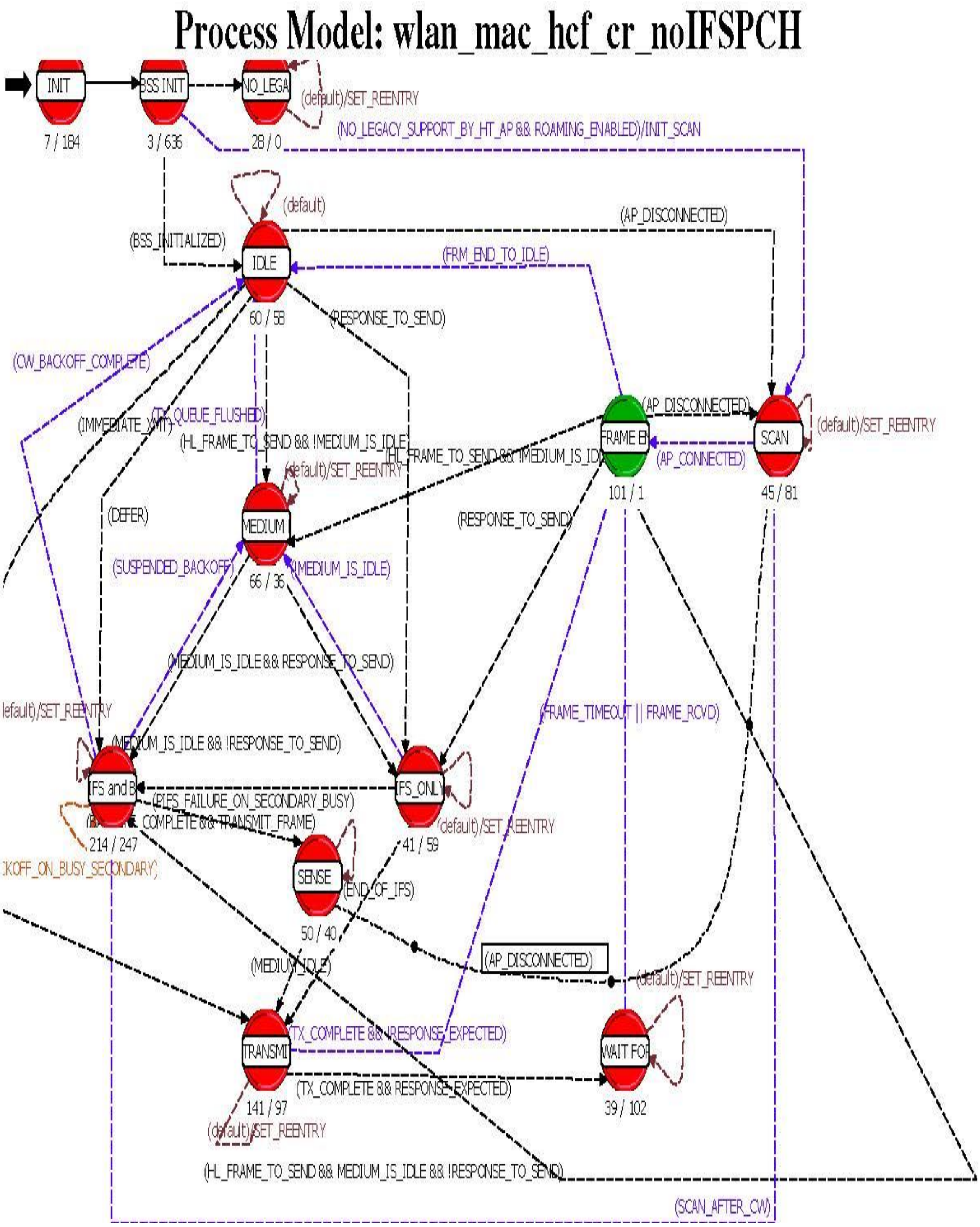
```
/* We enter into this state when we want to send a pakcet and  */
/* want to sense the channel before sending. The sense operation */


/* Make sure this is not a reentry into this state.              */
if (state_reentered == OPC_FALSE)
    {
    /* If we have been operating in 11p mode, then first switch     */
    /* back to the configured PHY technology.                       */
    /*if (phy_char_flag == WlanC_OFDM_11p)
        wlan_hcf_from_11p_phy_switch (); */

    /* Initiate the sensing procedure. Added by Nabil                               */
    wlan_sensing_cr ();

    /*set the duration of the sensing */
    sensing_duration = 0.05; /* 0.4 = 400 ms, 0.1 = 100 ms, 0.05 = 50ms   */


    /* probability affect by Nabil */

    wlan_sensingP_cr (0.95, 0.1);

    /* end of probability affect*/

    sensing_timeout_evh = op_intrpt_schedule_self (current_time + sensing_duration, WlanC_Sensing_Duration);

    /*CHECK IF THE PRIMARY OR SECONDARY CAHNNEL IS BUSY */
    if (wlan_flags->receiver_busy == OPC_TRUE || wlan_flags->secondary_ch_busy == OPC_TRUE)
    /* Set the flag indicating the ongoing sensing procedure.   Added by Nabil  */
       wlan_flags->sensing = OPC_TRUE;

     else {
           if (wlan_flags->receiver_busy == OPC_FALSE && wlan_flags->secondary_ch_busy == OPC_FALSE)

               wlan_flags->sensing = OPC_FALSE;
           }

    /* Record the state name for debugging purposes.             */
    if (debug_mode)
        {
        strcpy (current_state_name, "SENSING"); /*Changed from SCAN to SENSING by Nabil*/
        }
    }

#if defined (OPD_PARALLEL)
/* Unlock the mutex that serializes accessing the roaming related   */
/* information of this MAC.                                         */
op_prg_mt_mutex_unlock (roam_state_ptr->roam_info_mutex);
#endif
```

wlan_mac_hcf_cr_noIFSP.SENSE.Exit Executives

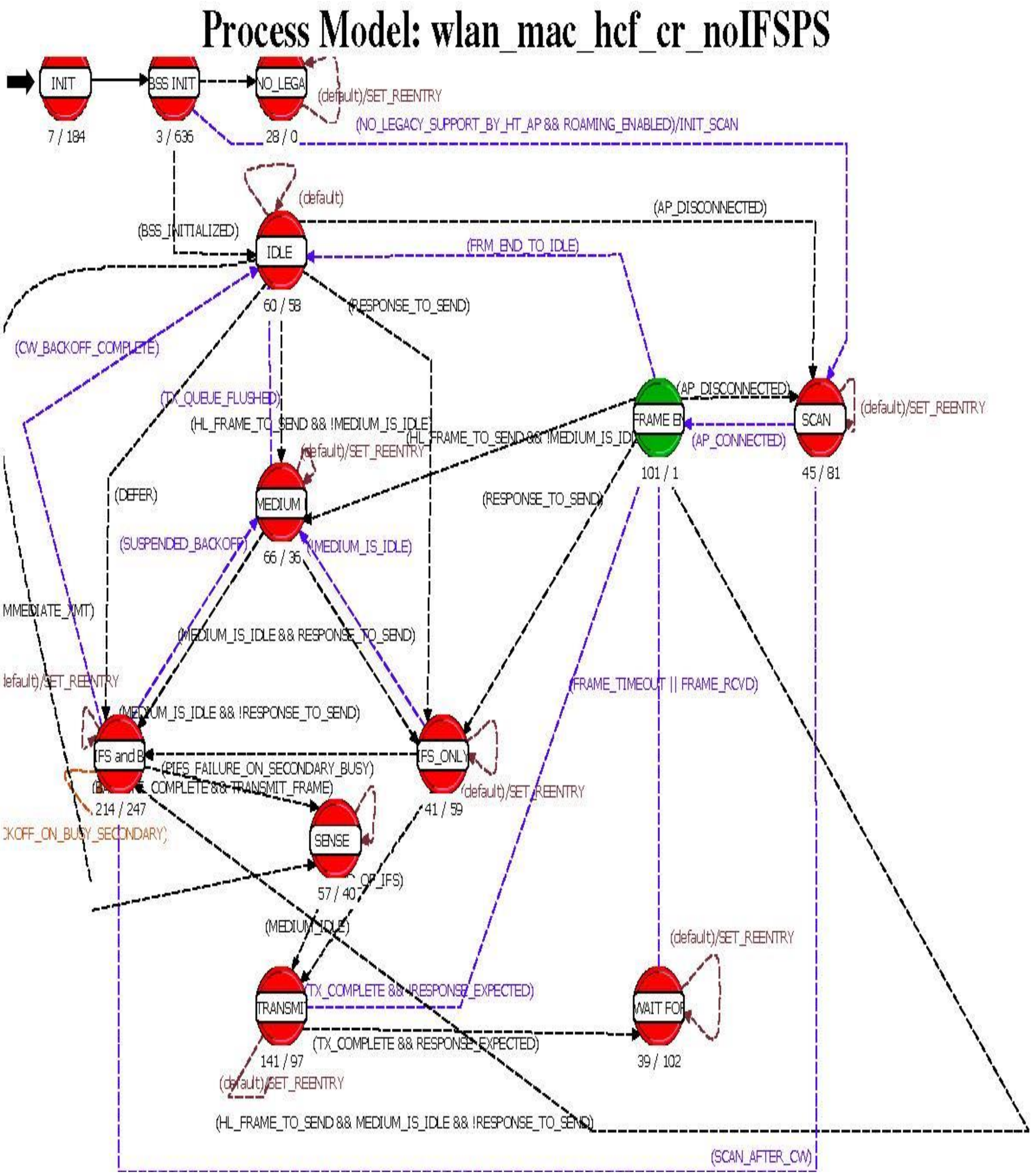
File Edit Options

```
#if defined (OPD_PARALLEL)
/* Lock the mutex that serializes accessing the roaming related     */
/* information of this MAC.                                          */
op_prg_mt_mutex_lock (roam_state_ptr->roam_info_mutex, 0);
#endif

/* Interrupt processing routine.                                     */
wlan_hcf_interrupts_process ();

/* Check whether the self interrupt set for the expiry of the sensing   */
/* period is delivered.     Added by Nabil */

if (wlan_flags->receiver_busy == OPC_FALSE)
{
op_ev_cancel_if_pending(sensing_timeout_evh); //added by Nabil
wlan_flags->sensing = OPC_FALSE;
}
else
    /* Reset the related flag since sensing is complete.       */
    {
    if (intrpt_type == OPC_INTRPT_SELF && intrpt_code == WlanC_Sensing_Duration)
      {
       op_ev_cancel_if_pending(sensing_timeout_evh); //added by Nabi
       wlan_flags->sensing = OPC_FALSE;  //changed from scanning to sensing by Nabil
       }
    }

if (!(intrpt_type == OPC_INTRPT_SELF && intrpt_code == WlanC_Sensing_Duration))
    {
    if (wlan_flags->sensing == OPC_TRUE)
      {

        wlan_sensing_cr ();
        /* probability affect by Nabil */

      //wlan_sensingP_cr (0.9, 0.1);

       /* end of probability affect*/
      }
    }
```

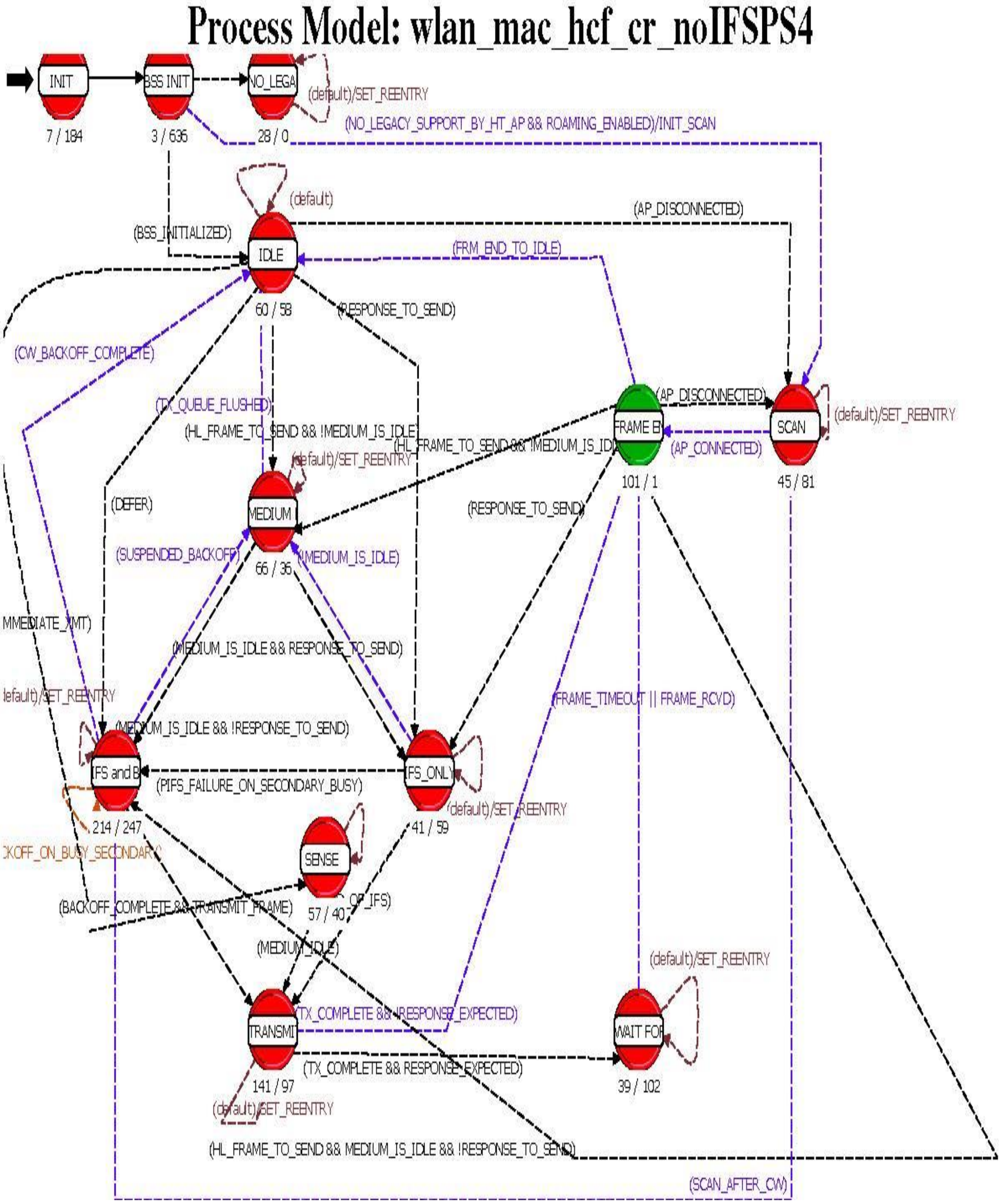
Appendix B 9 Layout for 'wlan_mac_hcf_cr_noIFSPCH' process model



Appendix B 10 Layout for 'wlan_mac_hcf_cr_noIFSPS' process model



Appendix B 11 Layout for 'wlan_mac_hcf_cr_noIFSPS4' process model



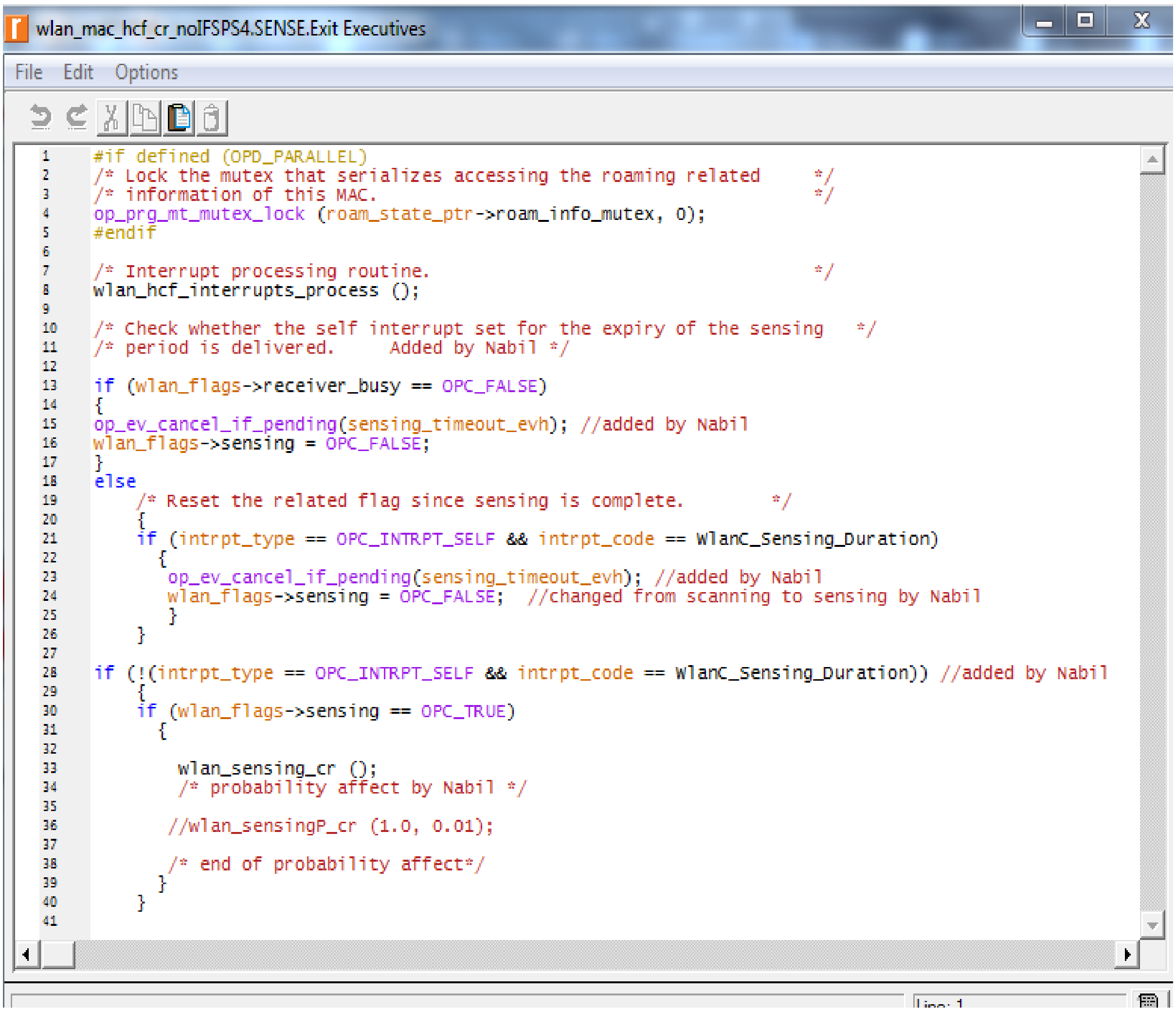
Appendix B 12 Sense state for 'wlan_mac_hcf_cr_noIFSPS4' and 'wlan_mac_hcf_cr_noIFSPS'

wlan_mac_hcf_cr_noIFSPS4.SENSE.Enter Executives

File Edit Options

```
/* We enter into this state when we want to send a pakcet and  */
/* want to sense the channel before sending. The sense operation */

/* Make sure this is not a reentry into this state.             */
if (state_reentered == OPC_FALSE)
    {

    /* Initiate the sensing procedure. Added by Nabil */
    wlan_sensing_cr ();

    /*set the duration of the sensing added by Nabil */
    //sensing_duration = 0.005; /* 0.4 = 400 ms, 0.1 = 100 ms, 0.05 = 50ms    */
    if (cur_tx_ac == WlanC_AC_VO)
            sensing_duration = 0.005;
        else if (cur_tx_ac == WlanC_AC_VI)
            sensing_duration = 0.01;
        else if (cur_tx_ac == WlanC_AC_BE)
            sensing_duration = 0.1;
        else
            sensing_duration = 0.3;

    /* probability affect by Nabil */
        if (cur_tx_ac == WlanC_AC_VO)
            wlan_sensingP_cr (0.9, 0.1);
        else if (cur_tx_ac == WlanC_AC_VI)
            wlan_sensingP_cr (0.95, 0.1);
        else if (cur_tx_ac == WlanC_AC_BE)
            wlan_sensingP_cr (0.95, 0.1);
        else
            wlan_sensingP_cr (0.95, 0.01);
   /* end of probability affect*/

    sensing_timeout_evh = op_intrpt_schedule_self (current_time + sensing_duration, Wl

    /*CHECK IF THE PRIMARY OR SECONDARY CAHNNEL IS BUSY */
    if (wlan_flags->receiver_busy == OPC_TRUE || wlan_flags->secondary_ch_busy == OPC_
    /* Set the flag indicating the ongoing sensing procedure.   Added by Nabil  */
       wlan_flags->sensing = OPC_TRUE;

     else {
           if (wlan_flags->receiver_busy == OPC_FALSE && wlan_flags->secondary_ch_busy

               wlan_flags->sensing = OPC_FALSE;
           }

    /* Record the state name for debugging purposes.             */
    if (debug_mode)
        {
        strcpy (current_state_name, "SENSING"); /*Changed from SCAN to SENSING by Nabi
        }
    }

#if defined (OPD_PARALLEL)
/* Unlock the mutex that serializes accessing the roaming related   */
/* information of this MAC.                                         */
op_prg_mt_mutex_unlock (roam_state_ptr->roam_info_mutex);
#endif
```

Line: 57



wlan_mac_hcf_cr_noIFSPS4.SENSE.Exit Executives

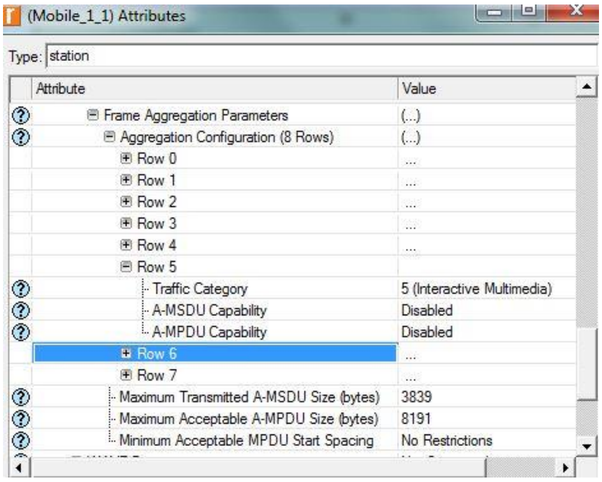
```
#if defined (OPD_PARALLEL)
/* Lock the mutex that serializes accessing the roaming related     */
/* information of this MAC.                                         */
op_prg_mt_mutex_lock (roam_state_ptr->roam_info_mutex, 0);
#endif

/* Interrupt processing routine.                                    */
wlan_hcf_interrupts_process ();

/* Check whether the self interrupt set for the expiry of the sensing   */
/* period is delivered.     Added by Nabil */

if (wlan_flags->receiver_busy == OPC_FALSE)
{
op_ev_cancel_if_pending(sensing_timeout_evh); //added by Nabil
wlan_flags->sensing = OPC_FALSE;
}
else
    /* Reset the related flag since sensing is complete.       */
    {
    if (intrpt_type == OPC_INTRPT_SELF && intrpt_code == WlanC_Sensing_Duration)
      {
       op_ev_cancel_if_pending(sensing_timeout_evh); //added by Nabil
       wlan_flags->sensing = OPC_FALSE;  //changed from scanning to sensing by Nabil
       }
    }

if (!(intrpt_type == OPC_INTRPT_SELF && intrpt_code == WlanC_Sensing_Duration)) //added by Nabil
    {
    if (wlan_flags->sensing == OPC_TRUE)
      {

        wlan_sensing_cr ();
        /* probability affect by Nabil */

       //wlan_sensingP_cr (1.0, 0.01);

       /* end of probability affect*/
      }
    }
```

Snapshot of settings for Section 5.5.1: Evaluations of the proposed sensing strategies

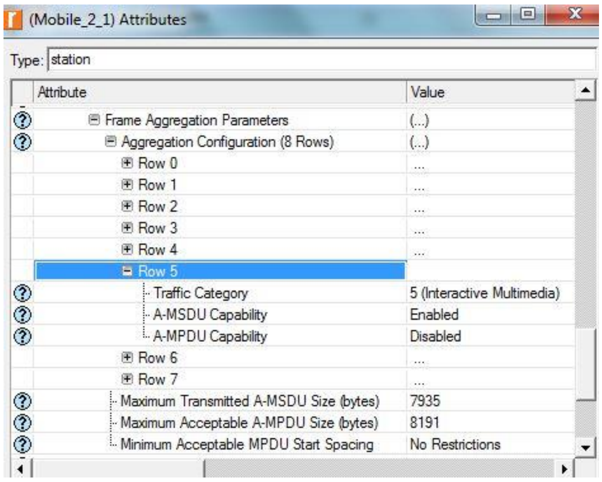
Appendix B 13 Network layout for comparing fixed sensing strategy with the proposed strategies

Snapshot of settings for Section 5.5.3: Impact of frame aggregation and sensing

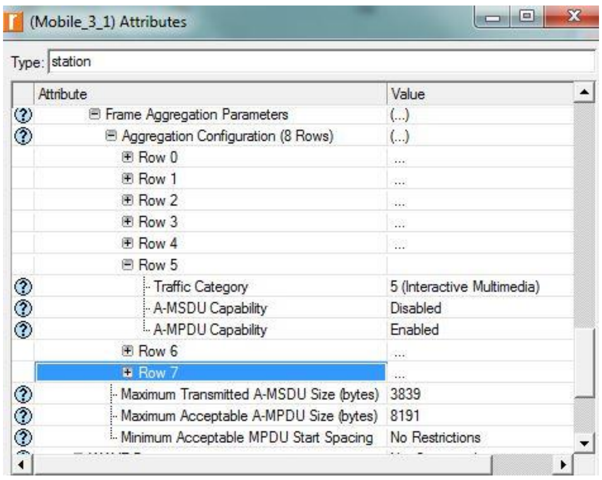
(Refer: attribute 'WLAN Parameters \ High Throughput Parameters \ Frame Aggregation Parameters')



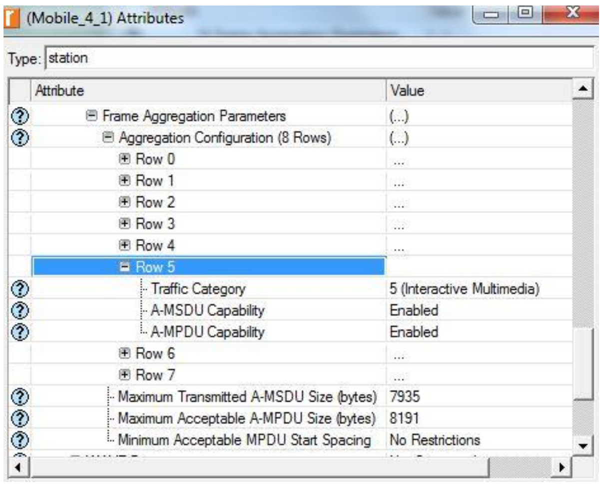
Frame aggregation configurations for BSS-0



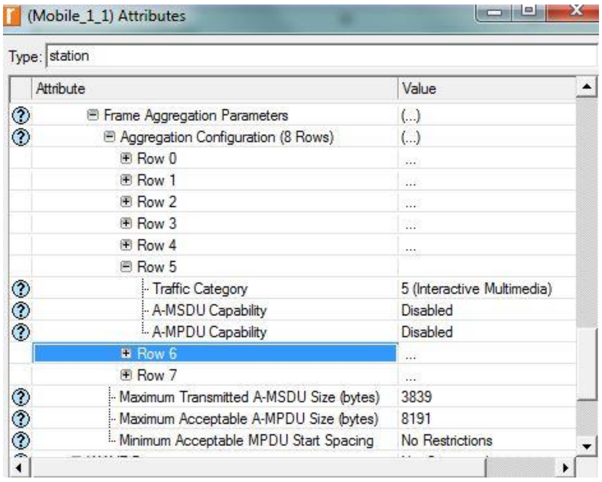
Frame aggregation configurations for BSS-1



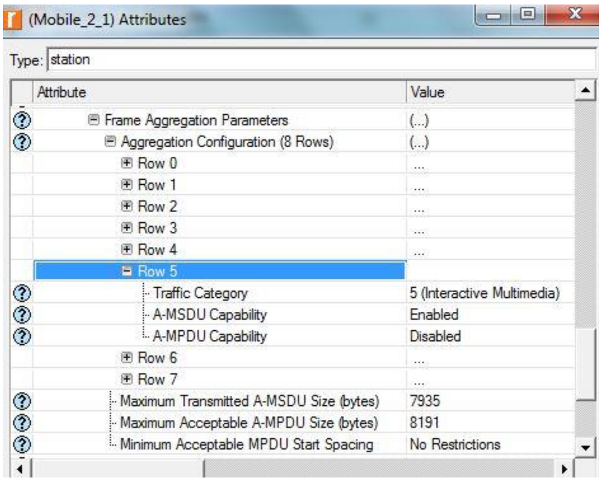
Frame aggregation configurations for BSS-2



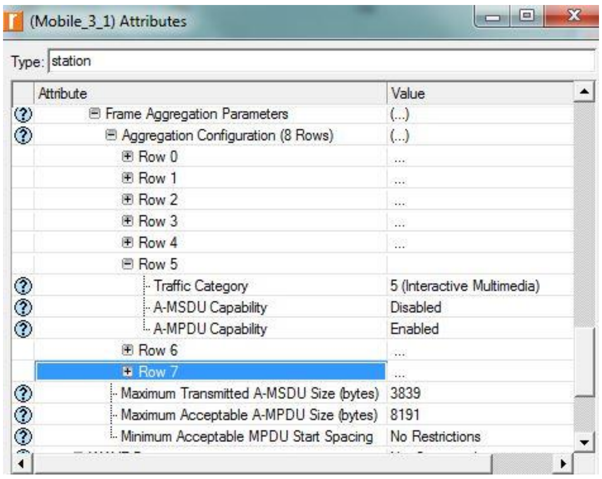
Frame aggregation configurations for BSS-3



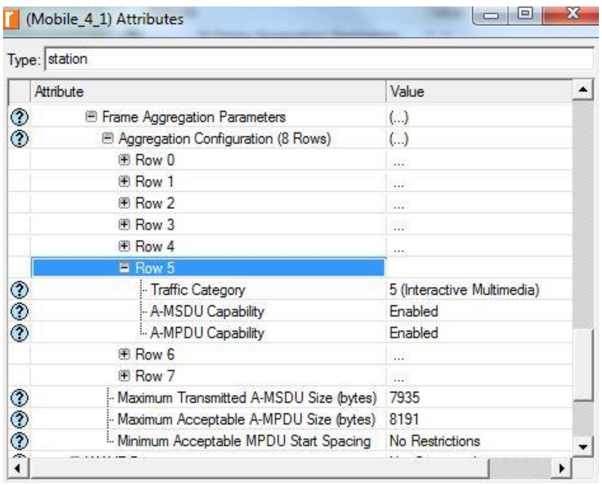
Frame aggregation configurations for BSS-0



Frame aggregation configurations for BSS-1

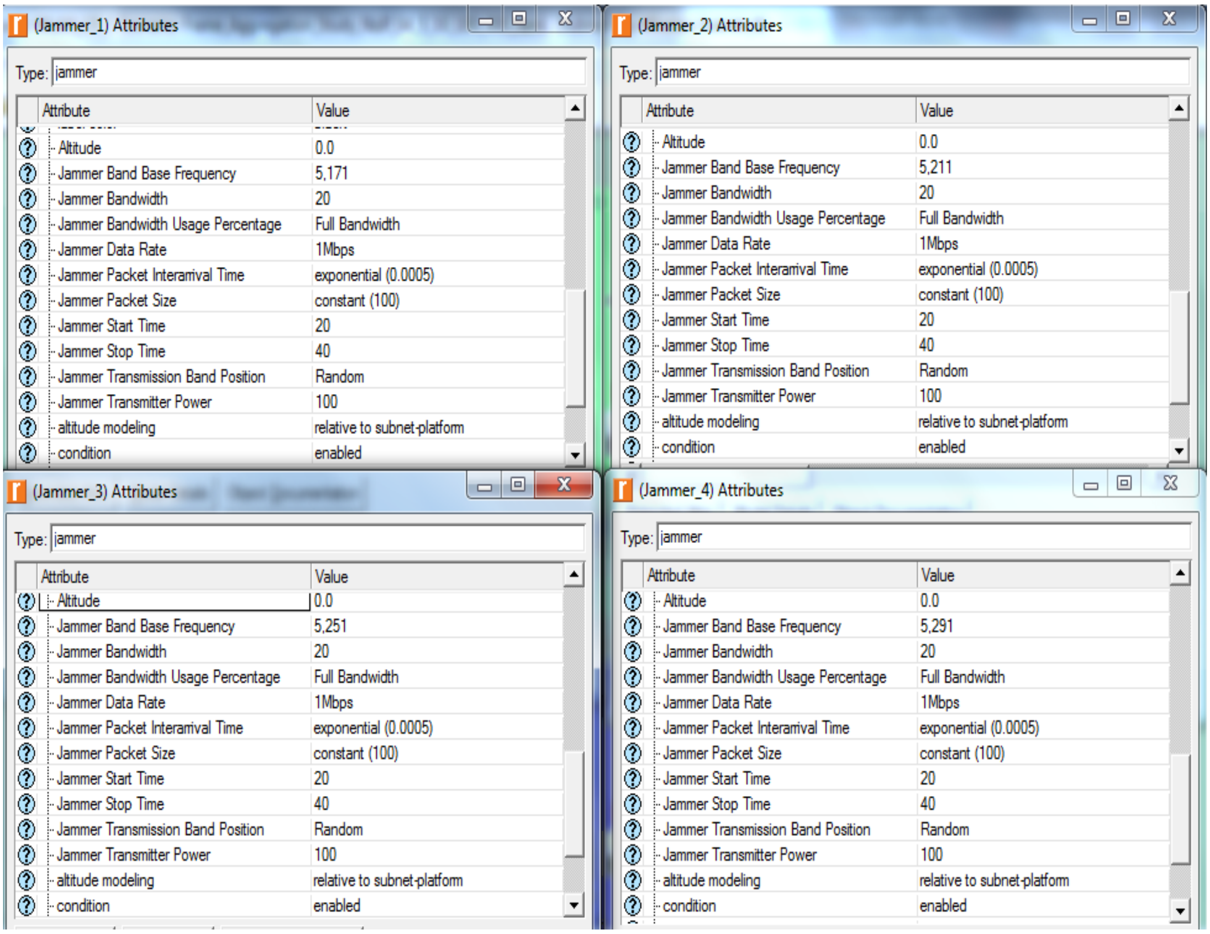


Frame aggregation configurations for BSS-2



Frame aggregation configurations for BSS-3

Appendix B 14 Attributes of the used four jammers



(Jammer_1) Attributes

Type: jammer

Attribute	Value
Altitude	0.0
Jammer Band Base Frequency	5,171
Jammer Bandwidth	20
Jammer Bandwidth Usage Percentage	Full Bandwidth
Jammer Data Rate	1Mbps
Jammer Packet Interarrival Time	exponential (0.0005)
Jammer Packet Size	constant (100)
Jammer Start Time	20
Jammer Stop Time	40
Jammer Transmission Band Position	Random
Jammer Transmitter Power	100
altitude modeling	relative to subnet-platform
condition	enabled

(Jammer_2) Attributes

Type: jammer

Attribute	Value
Altitude	0.0
Jammer Band Base Frequency	5,211
Jammer Bandwidth	20
Jammer Bandwidth Usage Percentage	Full Bandwidth
Jammer Data Rate	1Mbps
Jammer Packet Interarrival Time	exponential (0.0005)
Jammer Packet Size	constant (100)
Jammer Start Time	20
Jammer Stop Time	40
Jammer Transmission Band Position	Random
Jammer Transmitter Power	100
altitude modeling	relative to subnet-platform
condition	enabled

(Jammer_3) Attributes

Type: jammer

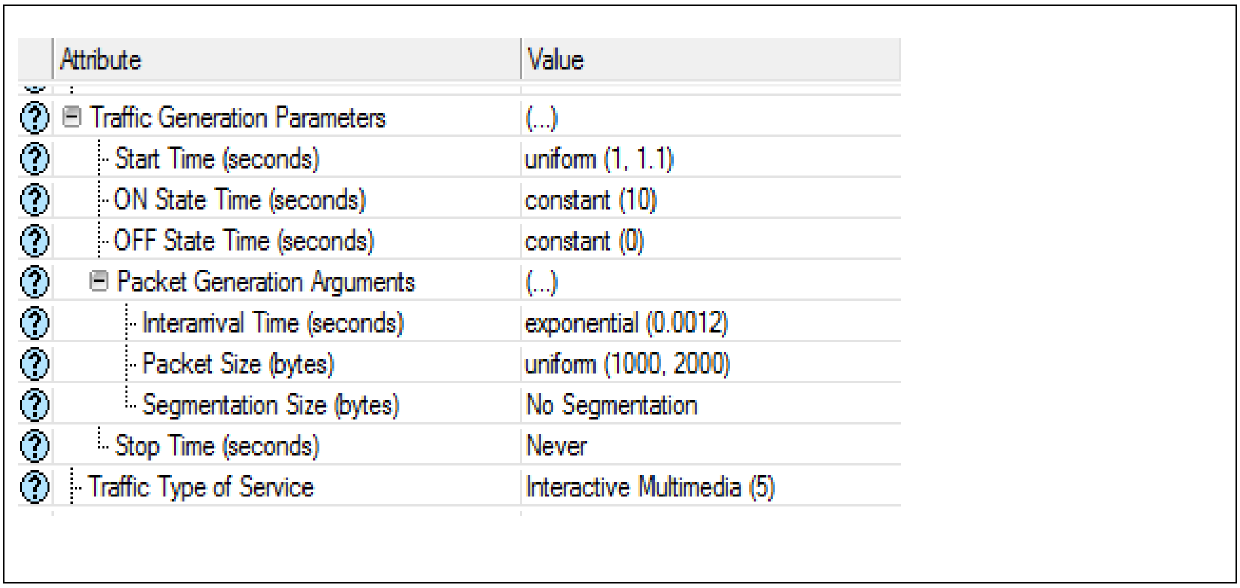
Attribute	Value
Altitude	0.0
Jammer Band Base Frequency	5,251
Jammer Bandwidth	20
Jammer Bandwidth Usage Percentage	Full Bandwidth
Jammer Data Rate	1Mbps
Jammer Packet Interarrival Time	exponential (0.0005)
Jammer Packet Size	constant (100)
Jammer Start Time	20
Jammer Stop Time	40
Jammer Transmission Band Position	Random
Jammer Transmitter Power	100
altitude modeling	relative to subnet-platform
condition	enabled

(Jammer_4) Attributes

Type: jammer

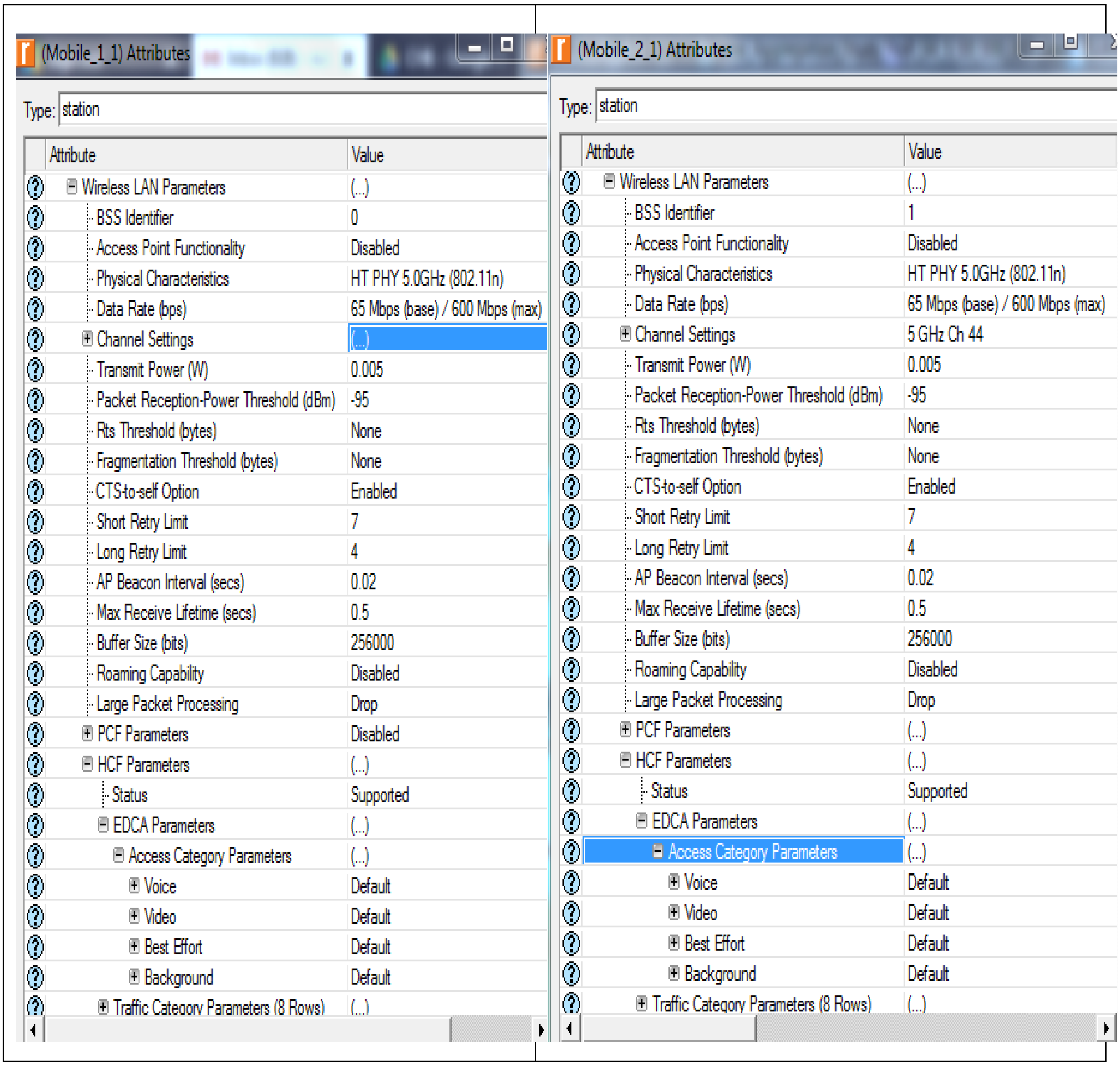
Attribute	Value
Altitude	0.0
Jammer Band Base Frequency	5,291
Jammer Bandwidth	20
Jammer Bandwidth Usage Percentage	Full Bandwidth
Jammer Data Rate	1Mbps
Jammer Packet Interarrival Time	exponential (0.0005)
Jammer Packet Size	constant (100)
Jammer Start Time	20
Jammer Stop Time	40
Jammer Transmission Band Position	Random
Jammer Transmitter Power	100
altitude modeling	relative to subnet-platform
condition	enabled

Appendix B 15 Attributes of the traffic used in Frame aggregation study



Attribute	Value
Traffic Generation Parameters	(...)
Start Time (seconds)	uniform (1, 1.1)
ON State Time (seconds)	constant (10)
OFF State Time (seconds)	constant (0)
Packet Generation Arguments	(...)
Interarrival Time (seconds)	exponential (0.0012)
Packet Size (bytes)	uniform (1000, 2000)
Segmentation Size (bytes)	No Segmentation
Stop Time (seconds)	Never
Traffic Type of Service	Interactive Multimedia (5)

Appendix B 16 Attributes of Mobile nodes used in each BSS of the Frame aggregation Study Scenario



(Mobile_1_1) Attributes

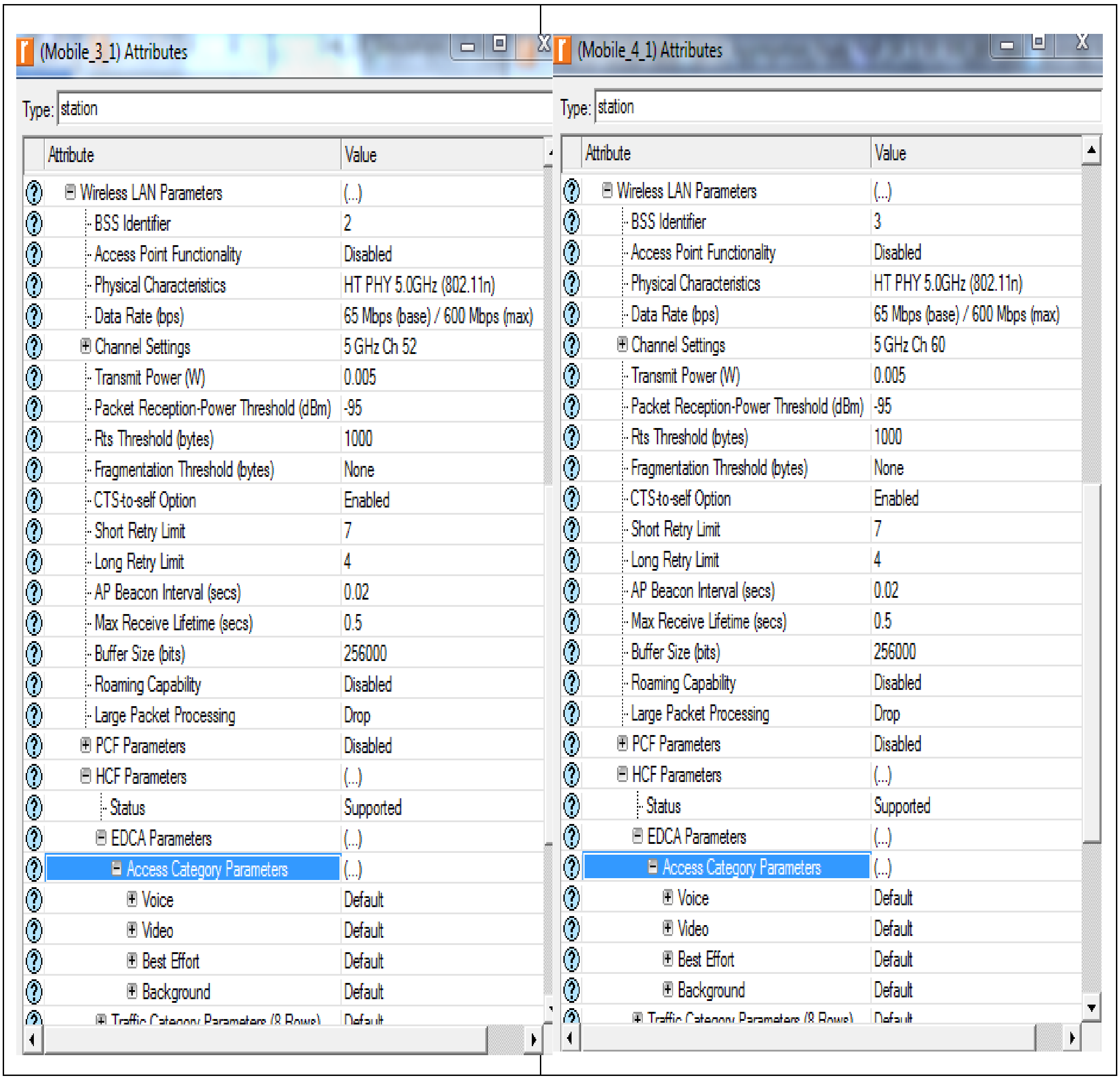
Type: station

Attribute	Value
Wireless LAN Parameters	(...)
BSS Identifier	0
Access Point Functionality	Disabled
Physical Characteristics	HT PHY 5.0GHz (802.11n)
Data Rate (bps)	65 Mbps (base) / 600 Mbps (max)
Channel Settings	(...)
Transmit Power (W)	0.005
Packet Reception-Power Threshold (dBm)	-95
Rts Threshold (bytes)	None
Fragmentation Threshold (bytes)	None
CTS-to-self Option	Enabled
Short Retry Limit	7
Long Retry Limit	4
AP Beacon Interval (secs)	0.02
Max Receive Lifetime (secs)	0.5
Buffer Size (bits)	256000
Roaming Capability	Disabled
Large Packet Processing	Drop
PCF Parameters	Disabled
HCF Parameters	(...)
Status	Supported
EDCA Parameters	(...)
Access Category Parameters	(...)
Voice	Default
Video	Default
Best Effort	Default
Background	Default
Traffic Category Parameters (8 Rows)	(...)

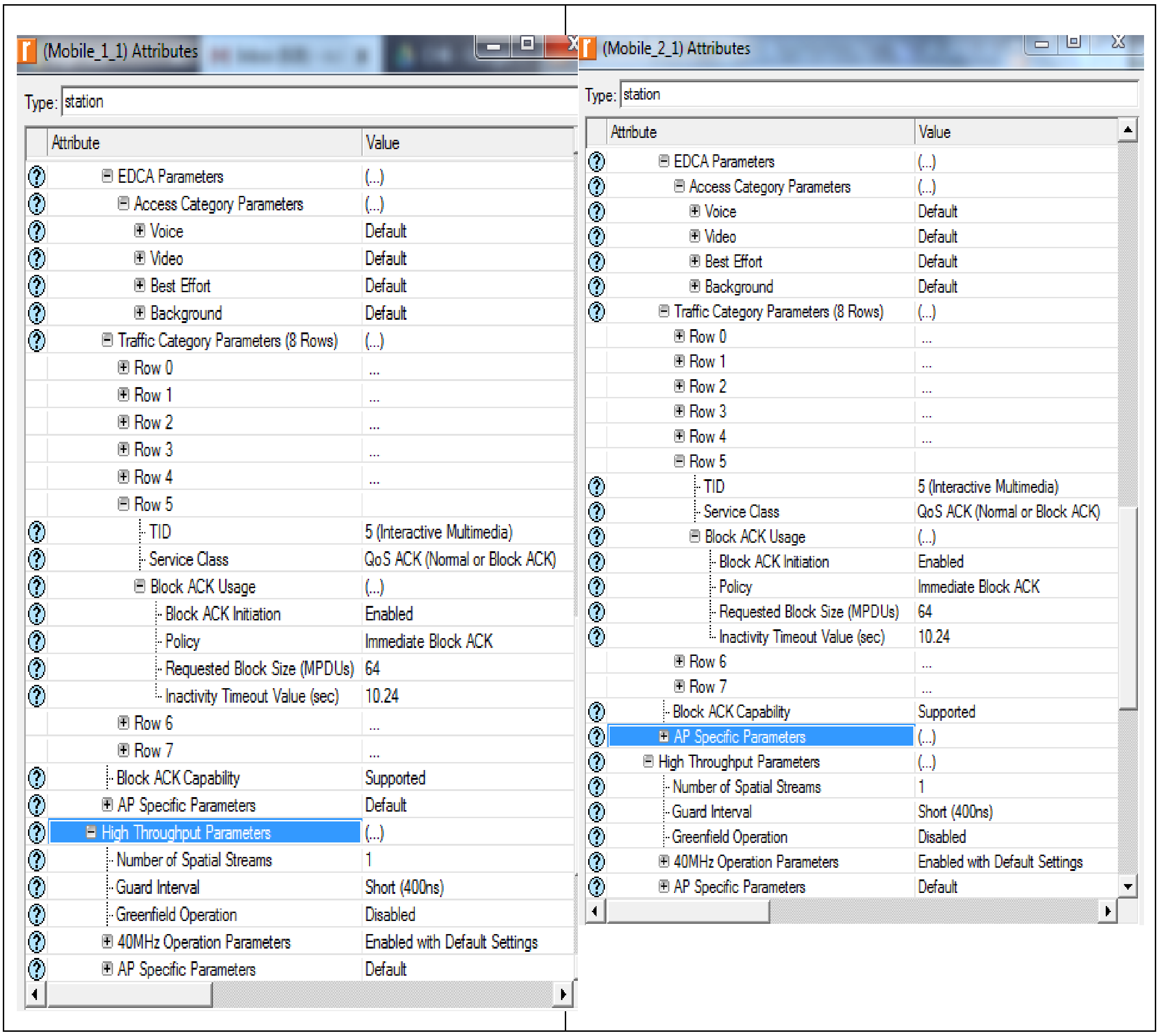
(Mobile_2_1) Attributes

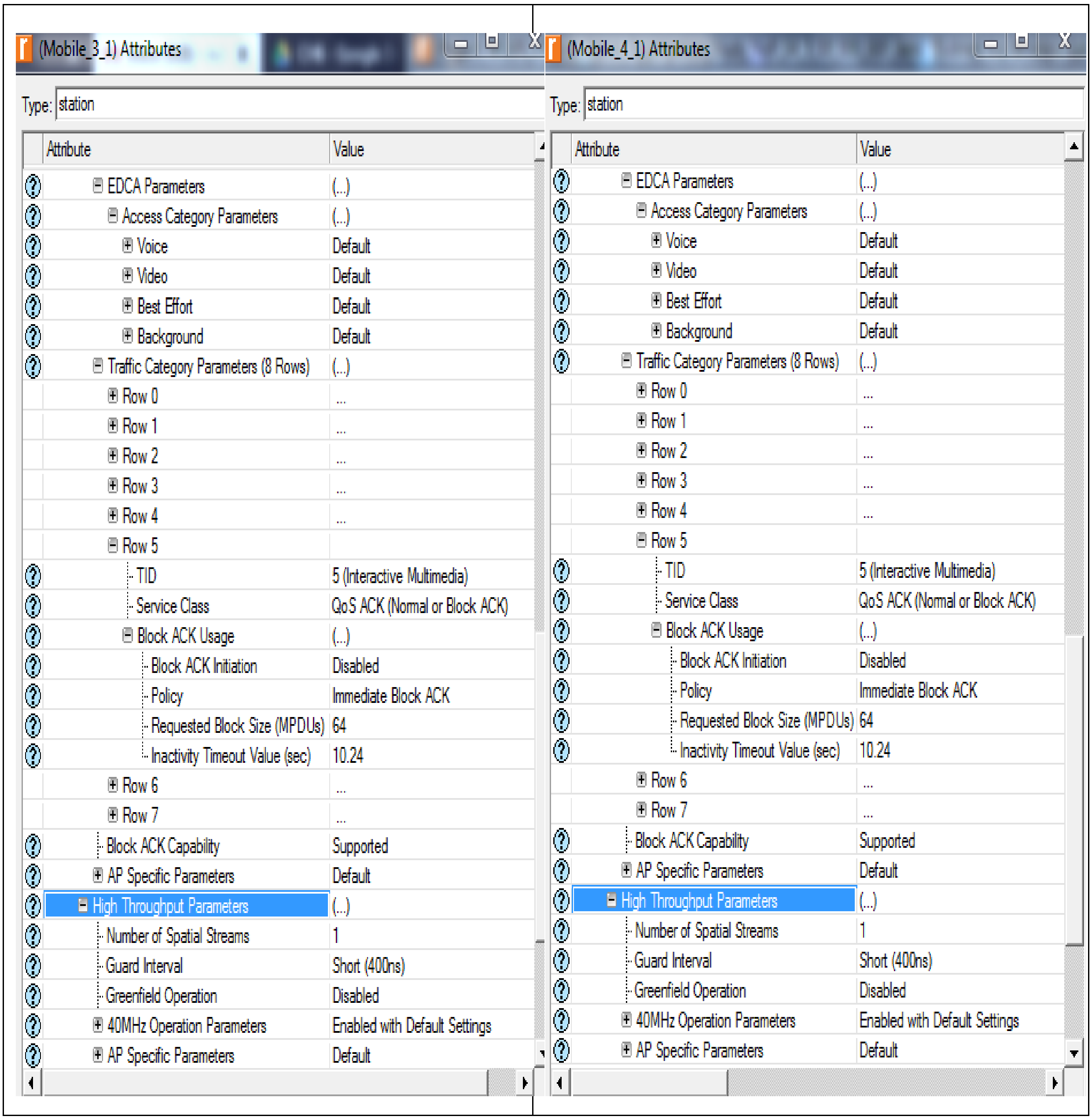
Type: station

Attribute	Value
Wireless LAN Parameters	(...)
BSS Identifier	1
Access Point Functionality	Disabled
Physical Characteristics	HT PHY 5.0GHz (802.11n)
Data Rate (bps)	65 Mbps (base) / 600 Mbps (max)
Channel Settings	5 GHz Ch 44
Transmit Power (W)	0.005
Packet Reception-Power Threshold (dBm)	-95
Rts Threshold (bytes)	None
Fragmentation Threshold (bytes)	None
CTS-to-self Option	Enabled
Short Retry Limit	7
Long Retry Limit	4
AP Beacon Interval (secs)	0.02
Max Receive Lifetime (secs)	0.5
Buffer Size (bits)	256000
Roaming Capability	Disabled
Large Packet Processing	Drop
PCF Parameters	(...)
HCF Parameters	(...)
Status	Supported
EDCA Parameters	(...)
Access Category Parameters	(...)
Voice	Default
Video	Default
Best Effort	Default
Background	Default
Traffic Category Parameters (8 Rows)	(...)



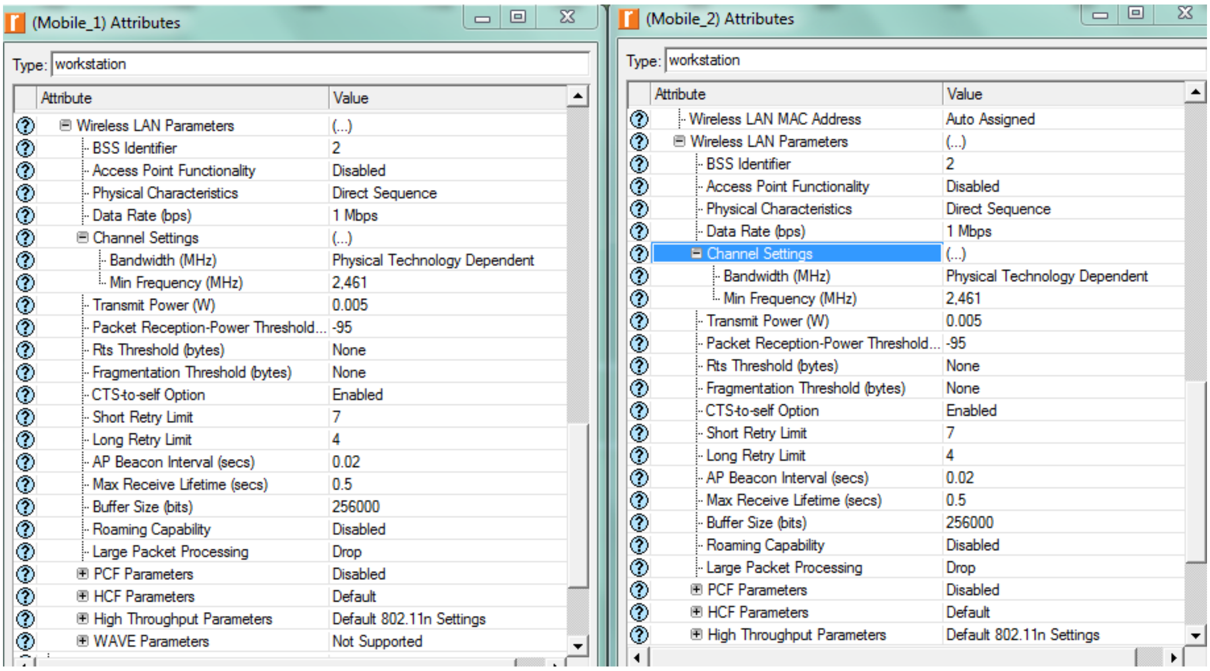
(Mobile_3_1) Attributes
Type: station
Attribute
Value
Wireless LAN Parameters
(...)
BSS Identifier
2
Access Point Functionality
Disabled
Physical Characteristics
HT PHY 5.0GHz (802.11n)
Data Rate (bps)
65 Mbps (base) / 600 Mbps (max)
Channel Settings
5 GHz Ch 52
Transmit Power (W)
0.005
Packet Reception-Power Threshold (dBm)
-95
Rts Threshold (bytes)
1000
Fragmentation Threshold (bytes)
None
CTS-to-self Option
Enabled
Short Retry Limit
7
Long Retry Limit
4
AP Beacon Interval (secs)
0.02
Max Receive Lifetime (secs)
0.5
Buffer Size (bits)
256000
Roaming Capability
Disabled
Large Packet Processing
Drop
PCF Parameters
Disabled
HCF Parameters
(...)
Status
Supported
EDCA Parameters
(...)
Access Category Parameters
(...)
Voice
Default
Video
Default
Best Effort
Default
Background
Default
(Mobile_4_1) Attributes
Type: station
Attribute
Value
Wireless LAN Parameters
(...)
BSS Identifier
3
Access Point Functionality
Disabled
Physical Characteristics
HT PHY 5.0GHz (802.11n)
Data Rate (bps)
65 Mbps (base) / 600 Mbps (max)
Channel Settings
5 GHz Ch 60
Transmit Power (W)
0.005
Packet Reception-Power Threshold (dBm)
-95
Rts Threshold (bytes)
1000
Fragmentation Threshold (bytes)
None
CTS-to-self Option
Enabled
Short Retry Limit
7
Long Retry Limit
4
AP Beacon Interval (secs)
0.02
Max Receive Lifetime (secs)
0.5
Buffer Size (bits)
256000
Roaming Capability
Disabled
Large Packet Processing
Drop
PCF Parameters
Disabled
HCF Parameters
(...)
Status
Supported
EDCA Parameters
(...)
Access Category Parameters
(...)
Voice
Default
Video
Default
Best Effort
Default
Background
Default


(Mobile_1_1) Attributes
Type: station
Attribute
Value
EDCA Parameters
(...)
Access Category Parameters
(...)
Voice
Default
Video
Default
Best Effort
Default
Background
Default
Traffic Category Parameters (8 Rows)
(...)
Row 0
Row 1
Row 2
Row 3
Row 4
Row 5
TID
5 (Interactive Multimedia)
Service Class
QoS ACK (Normal or Block ACK)
Block ACK Usage
(...)
Block ACK Initiation
Enabled
Policy
Immediate Block ACK
Requested Block Size (MPDUs)
64
Inactivity Timeout Value (sec)
10.24
Row 6
Row 7
Block ACK Capability
Supported
AP Specific Parameters
Default
High Throughput Parameters
(...)
Number of Spatial Streams
1
Guard Interval
Short (400ns)
Greenfield Operation
Disabled
40MHz Operation Parameters
Enabled with Default Settings
AP Specific Parameters
Default
(Mobile_2_1) Attributes
Type: station
Attribute
Value
EDCA Parameters
(...)
Access Category Parameters
(...)
Voice
Default
Video
Default
Best Effort
Default
Background
Default
Traffic Category Parameters (8 Rows)
(...)
Row 0
Row 1
Row 2
Row 3
Row 4
Row 5
TID
5 (Interactive Multimedia)
Service Class
QoS ACK (Normal or Block ACK)
Block ACK Usage
(...)
Block ACK Initiation
Enabled
Policy
Immediate Block ACK
Requested Block Size (MPDUs)
64
Inactivity Timeout Value (sec)
10.24
Row 6
Row 7
Block ACK Capability
Supported
AP Specific Parameters
(...)
High Throughput Parameters
(...)
Number of Spatial Streams
1
Guard Interval
Short (400ns)
Greenfield Operation
Disabled
40MHz Operation Parameters
Enabled with Default Settings
AP Specific Parameters
Default

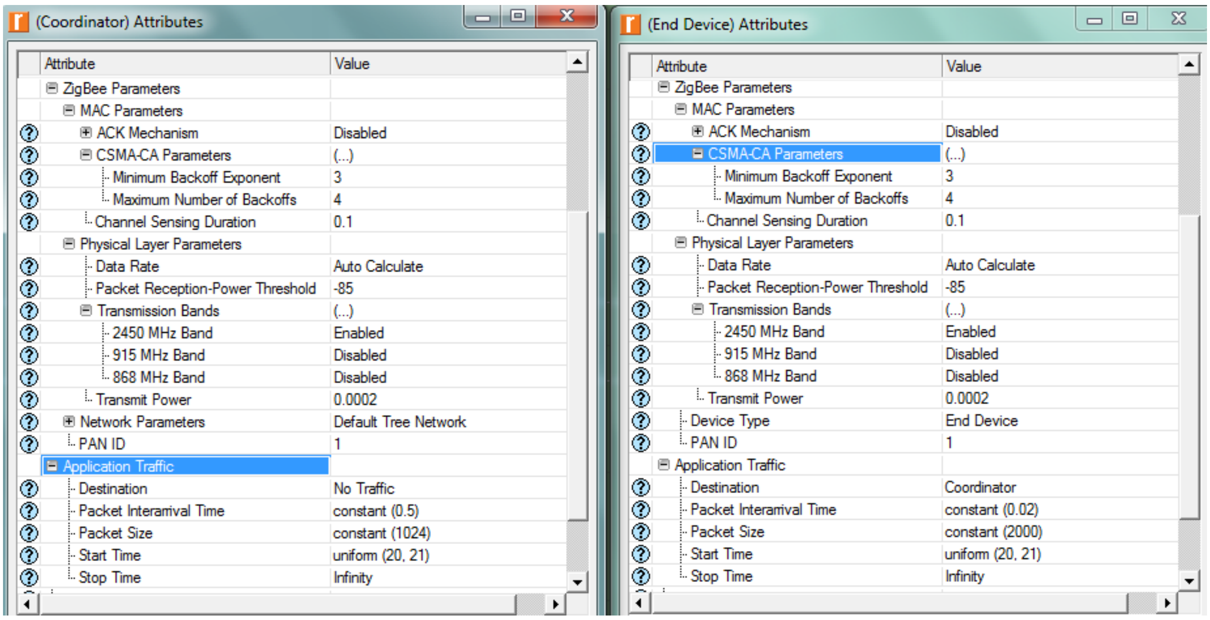

(Mobile_3_1) Attributes
Type: station
Attribute
Value
EDCA Parameters (...)
Access Category Parameters (...)
Voice Default
Video Default
Best Effort Default
Background Default
Traffic Category Parameters (8 Rows) (...)
Row 0 ...
Row 1 ...
Row 2 ...
Row 3 ...
Row 4 ...
Row 5
TID 5 (Interactive Multimedia)
Service Class QoS ACK (Normal or Block ACK)
Block ACK Usage (...)
Block ACK Initiation Disabled
Policy Immediate Block ACK
Requested Block Size (MPDUs) 64
Inactivity Timeout Value (sec) 10.24
Row 6 ...
Row 7 ...
Block ACK Capability Supported
AP Specific Parameters Default
High Throughput Parameters (...)
Number of Spatial Streams 1
Guard Interval Short (400ns)
Greenfield Operation Disabled
40MHz Operation Parameters Enabled with Default Settings
AP Specific Parameters Default
(Mobile_4_1) Attributes
Type: station
Attribute
Value
EDCA Parameters (...)
Access Category Parameters (...)
Voice Default
Video Default
Best Effort Default
Background Default
Traffic Category Parameters (8 Rows) (...)
Row 0 ...
Row 1 ...
Row 2 ...
Row 3 ...
Row 4 ...
Row 5
TID 5 (Interactive Multimedia)
Service Class QoS ACK (Normal or Block ACK)
Block ACK Usage (...)
Block ACK Initiation Disabled
Policy Immediate Block ACK
Requested Block Size (MPDUs) 64
Inactivity Timeout Value (sec) 10.24
Row 6 ...
Row 7 ...
Block ACK Capability Supported
AP Specific Parameters Default
High Throughput Parameters (...)
Number of Spatial Streams 1
Guard Interval Short (400ns)
Greenfield Operation Disabled
40MHz Operation Parameters Enabled with Default Settings
AP Specific Parameters Default

Snapshots of settings for Section 5.5.4: Impact of coexistence

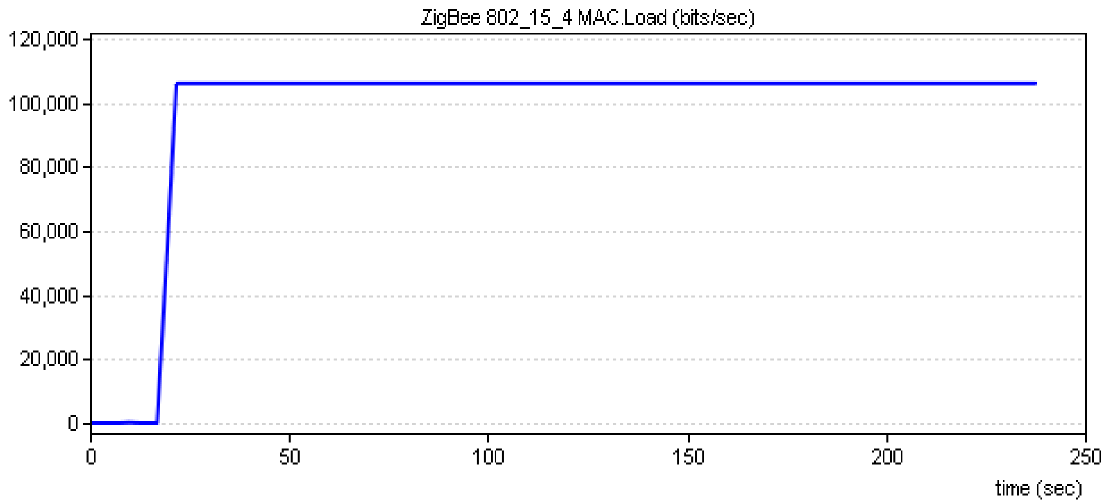
Appendix B 17 Attributes of mobile (QACR-MAC) nodes



Appendix B 18 Attributes of ZigBee nodes

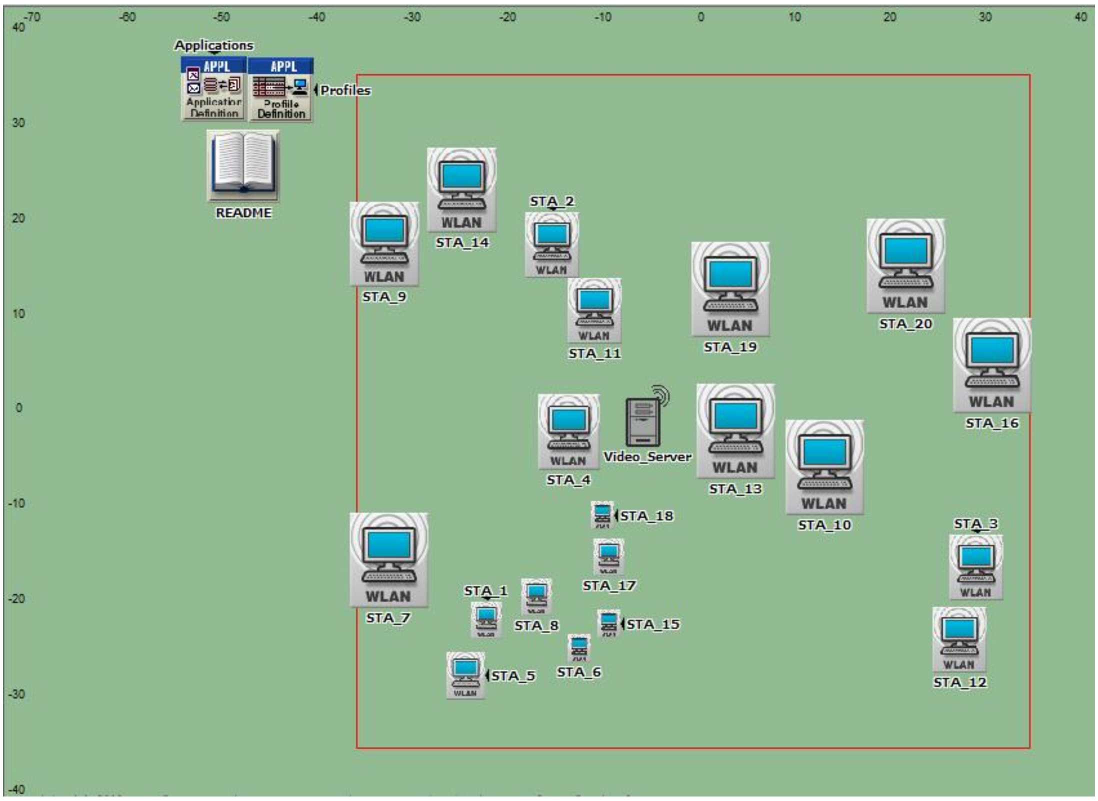


Appendix B 19 Load traffic from End device to Coordinator in ZigBee network



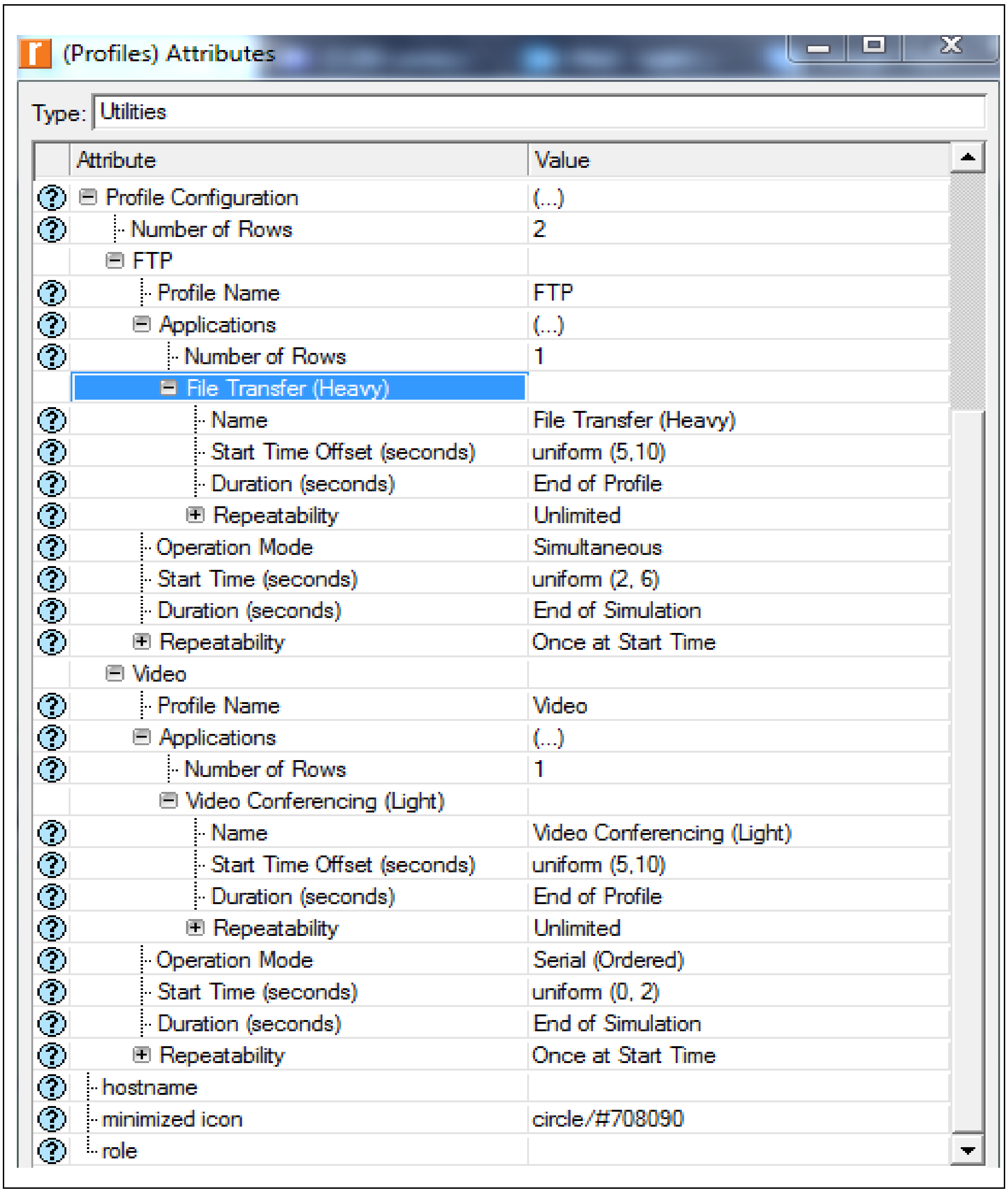
Snapshots of settings for Section 5.5.5: Impact of CR nodes number

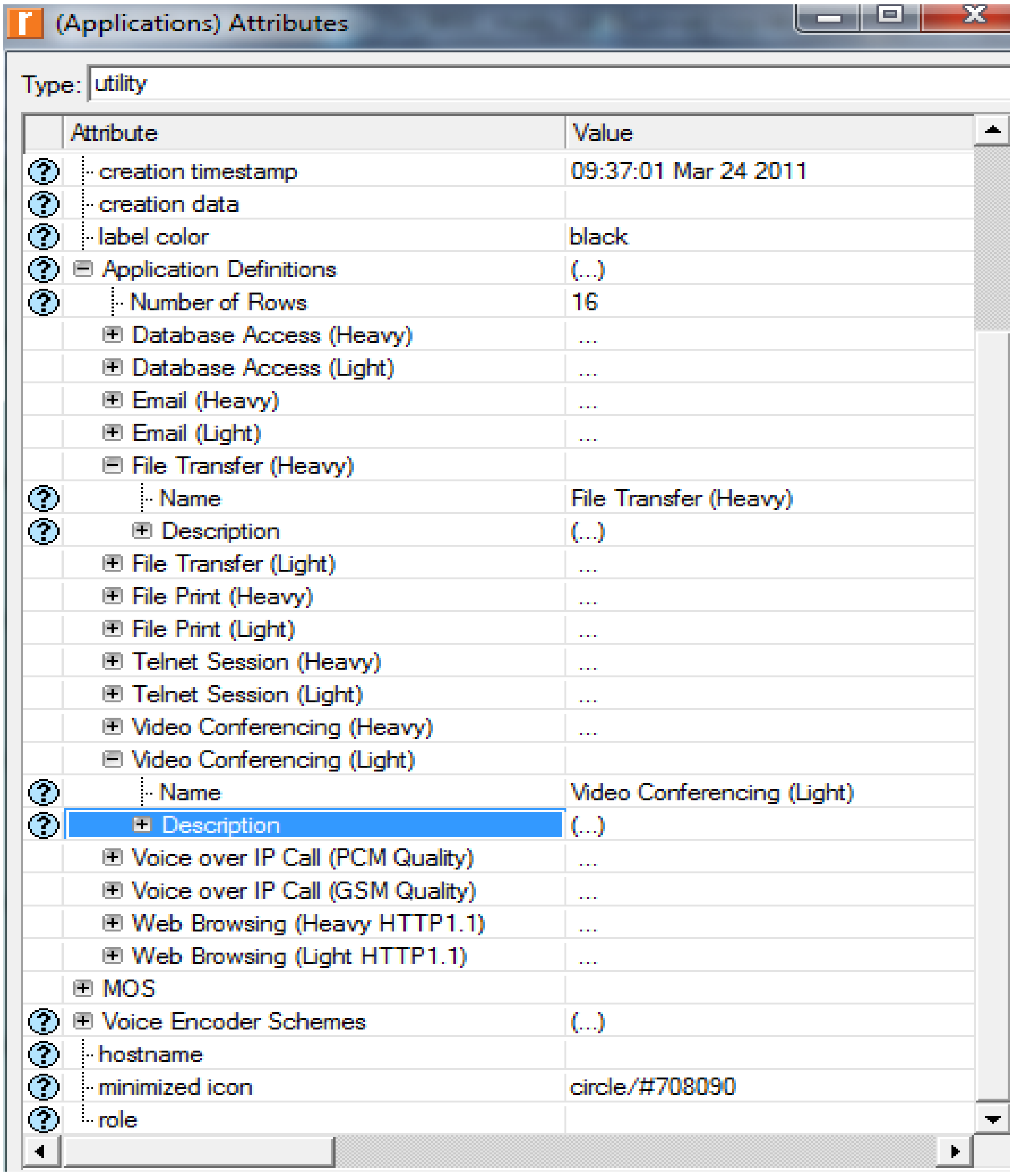
Appendix B 20 Network layout for studying the CR nodes number impact when all nodes are active (scale in meters)



Appendix B 21 Applications profile

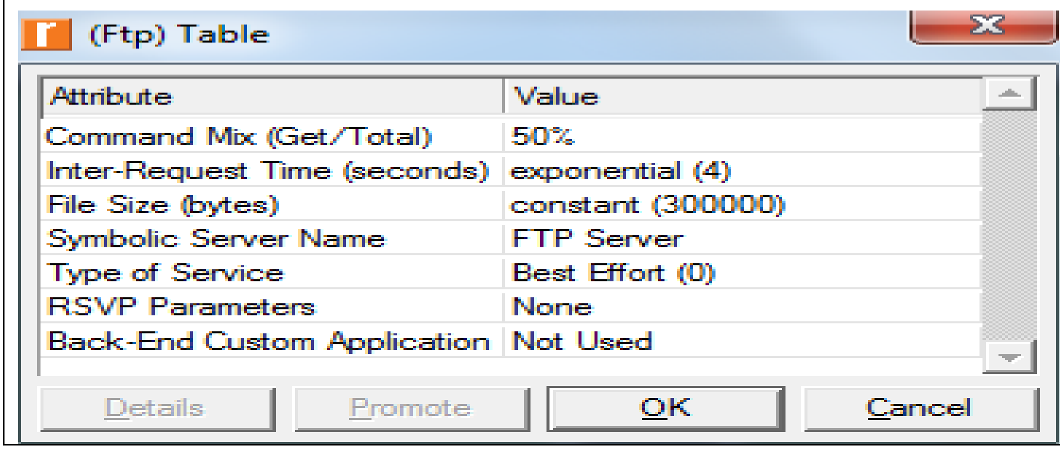
Attribute	Value
Wireless LAN	
Wireless LAN MAC Address	Auto Assigned
Wireless LAN Parameters	(...)
BSS Identifier	0
Access Point Functionality	Disabled
Physical Characteristics	HT PHY 2.4GHz (802.11n)
Data Rate (bps)	65 Mbps (base) / 600 Mbps (max)
Channel Settings	Auto Assigned
Transmit Power (W)	0.1
Packet Reception-Power Threshold (dBm)	-95
Rts Threshold (bytes)	None
Fragmentation Threshold (bytes)	None
CTS-to-self Option	Enabled
Short Retry Limit	7
Long Retry Limit	4
AP Beacon Interval (secs)	0.02
Max Receive Lifetime (secs)	0.5
Buffer Size (bits)	256000
Roaming Capability	Disabled
Large Packet Processing	Drop
PCF Parameters	Disabled
HCF Parameters	Default
High Throughput Parameters	(...)
Number of Spatial Streams	promoted
Guard Interval	promoted
Greenfield Operation	promoted
40MHz Operation Parameters	Disabled
AP Specific Parameters	Default
Frame Aggregation Parameters	(...)
Aggregation Configuration (8 Rows)	Disabled for All Traffic Categories


(Profiles) Attributes
Type: Utilities
Attribute
Value
Profile Configuration
(...)
Number of Rows
2
FTP
Profile Name
FTP
Applications
(...)
Number of Rows
1
File Transfer (Heavy)
Name
File Transfer (Heavy)
Start Time Offset (seconds)
uniform (5,10)
Duration (seconds)
End of Profile
Repeatability
Unlimited
Operation Mode
Simultaneous
Start Time (seconds)
uniform (2, 6)
Duration (seconds)
End of Simulation
Repeatability
Once at Start Time
Video
Profile Name
Video
Applications
(...)
Number of Rows
1
Video Conferencing (Light)
Name
Video Conferencing (Light)
Start Time Offset (seconds)
uniform (5,10)
Duration (seconds)
End of Profile
Repeatability
Unlimited
Operation Mode
Serial (Ordered)
Start Time (seconds)
uniform (0, 2)
Duration (seconds)
End of Simulation
Repeatability
Once at Start Time
hostname
minimized icon
circle/#708090
role


(Applications) Attributes

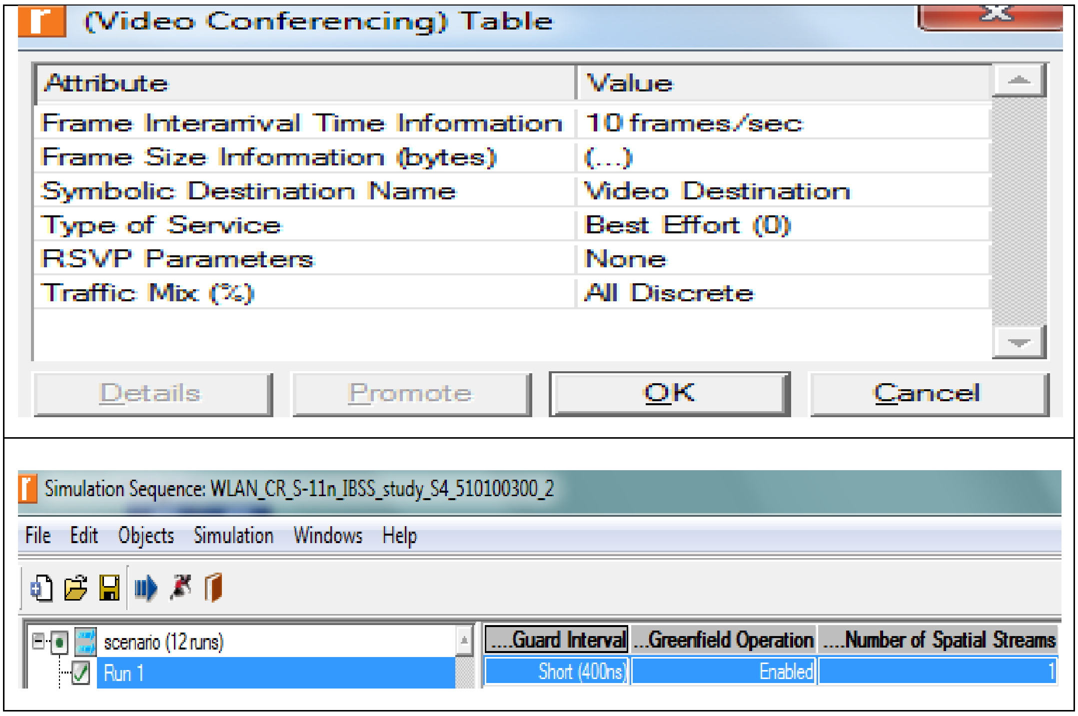
Type: utility

Attribute	Value
creation timestamp	09:37:01 Mar 24 2011
creation data	
label color	black
Application Definitions	(...)
Number of Rows	16
Database Access (Heavy)	...
Database Access (Light)	...
Email (Heavy)	...
Email (Light)	...
File Transfer (Heavy)	
Name	File Transfer (Heavy)
Description	(...)
File Transfer (Light)	...
File Print (Heavy)	...
File Print (Light)	...
Telnet Session (Heavy)	...
Telnet Session (Light)	...
Video Conferencing (Heavy)	...
Video Conferencing (Light)	
Name	Video Conferencing (Light)
Description	(...)
Voice over IP Call (PCM Quality)	...
Voice over IP Call (GSM Quality)	...
Web Browsing (Heavy HTTP1.1)	...
Web Browsing (Light HTTP1.1)	...
MOS	
Voice Encoder Schemes	(...)
hostname	
minimized icon	circle/#708090
role	


(Ftp) Table

Attribute	Value
Command Mix (Get/Total)	50%
Inter-Request Time (seconds)	exponential (4)
File Size (bytes)	constant (300000)
Symbolic Server Name	FTP Server
Type of Service	Best Effort (0)
RSVP Parameters	None
Back-End Custom Application	Not Used

Details Promote OK Cancel


(Video Conferencing) Table
Attribute
Value
Frame Interarrival Time Information
10 frames/sec
Frame Size Information (bytes)
(...)
Symbolic Destination Name
Video Destination
Type of Service
Best Effort (0)
RSVP Parameters
None
Traffic Mix (%)
All Discrete
Details
Promote
OK
Cancel
Simulation Sequence: WLAN_CR_S-11n_IBSS_study_S4_510100300_2
File
Edit
Objects
Simulation
Windows
Help
scenario (12 runs)
Run 1
....Guard Interval
...Greenfield Operation
....Number of Spatial Streams
Short (400ns)
Enabled
1